

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV AUTOMATIZACE A INFORMATIKY

FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

ZPRACOVÁNÍ NEURČITÝCH ÚDAJŮ V DATABÁZÍCH

PROCESSING OF UNCERTAIN INFORMATION IN DATABASES

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. PETR MORÁVEK

VEDOUCÍ PRÁCE
SUPERVISOR

doc. RNDr. Ing. MILOŠ ŠEDA, Ph.D.

BRNO 2009

Vysoké učení technické v Brně, Fakulta strojního inženýrství

Ústav automatizace a informatiky

Akademický rok: 2008/2009

ZADÁNÍ DIPLOMOVÉ PRÁCE

student(ka): Bc. Petr Morávek

který/která studuje v **magisterském navazujícím studijním programu**

obor: **Aplikovaná informatika a řízení (3902T001)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

Zpracování neurčitých údajů v databázích

v anglickém jazyce:

Processing of Uncertain Information in Databases

Stručná charakteristika problematiky úkolu:

V klasických databázích je obvyklé, že data, která se vztahují k entitám databáze, jsou popsána deterministicky, v krajním případě může některý z údajů chybět. V praxi však sledovaný údaj může být neurčitý a lze jej charakterizovat např. prostředky fuzzy množin nebo pravděpodobnostní algebry. Diplomová práce se zaměří na popis a zpracování neurčitých údajů včetně implementace některého z možných přístupů.

Cíle diplomové práce:

Cílem je provést klasifikaci datových modelů pro zpracování neurčitých informací, popsat databázové operace ve zvoleném modelu a způsob vyhodnocování odezvy na dotazy obsahující i jazykové kvalifikátory a popsané metody implementovat.

Seznam odborné literatury:

- [1] Bosc, P. and Kacprzyk, J. (eds.): Fuzziness in Database Management Systems. Physica-Verlag, Heidelberg, 1995.
- [2] Petry, F.E.: Fuzzy Databases. Principles and Applications. Kluwer Academic Publishers, Boston, 1996.
- [3] Šeda, M. and Dvořák, J.: A Framework for Multicriteria Selection Based on Measuring Query Responses. In Proceedings of the 3rd International Conference in Fuzzy Logic and Technology EUSFLAT 2003. European Society for Fuzzy Logic and Technology, Zittau (Germany), 2003, pp. 445-450.
- [4] Yazici, A. and George, R.: Fuzzy Database Modeling. Physica-Verlag, Heildeberg, 1999.

Vedoucí diplomové práce: doc. RNDr. Ing. Miloš Šeda, Ph.D.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2008/2009.

V Brně, dne

L.S.

doc. RNDr. Ing. Miloš Šeda, Ph.D.
Ředitel ústavu

doc. RNDr. Miroslav Doupovec, CSc.
Děkan fakulty

ABSTRAKT

Následující diplomová práce se zabývá zpracováním neurčitých údajů v databázích. Jako neurčité údaje jsou zvoleny vágní požadavky zákazníků při výběru notebooků během nákupu v klasických obchodech. Snahou této práce je vytvořit moderní aplikaci e-shopu notebooků doplněnou o expertní fuzzy systém, který má za úkol pomoci při výběru notebooků v internetovém obchodě zákazníkům, nemajícím dostatečné informace o technických parametrech a aktuálních trendech notebooků.

ABSTRACT

The following diploma thesis focuses on processing of uncertain information in databases. Uncertain information represents vague customer requests during laptop choice in classic shop purchasing. Effort of the work is to develop a modern e-shop application selling laptops, which is based on expert fuzzy system helping customers to choose a laptop without knowledge of technical specifications and current trends.

KLÍČOVÁ SLOVA

Fuzzy logika, neurčitost, vágnost, lingvistické proměnné, funkce příslušnosti, www aplikace, e-shop, expertní fuzzy systém, databáze, PHP, SQL, AJAX, administrační systém, skladový systém, fórum.

KEYWORDS

Fuzzy Logic, uncertainty, vagueness, linguistic variables, membership functions, web applications, e-shop, fuzzy expert system, database, PHP, SQL, AJAX, administration system, storage system, a forum.

PODĚKOVÁNÍ

Děkuji vedoucímu diplomové práce Doc. RNDr. Ing. Miloši Šedovi Ph.D. za výborné konzultace, vstřícnou pomoc a věcné připomínky k zdokonalení diplomové práce. Dále bych chtěl poděkovat rodičům za podporu při studiu a v neposlední řadě Bc. Romanu Kabelkovi za vstřícnou technickou podporu.

V Brně 17. 5. 2009

OBSAH

ZADÁNÍ DIPLOMOVÉ PRÁCE.....	3
ABSTRAKT	5
PODĚKOVÁNÍ.....	7
OBSAH	9
1 ÚVOD.....	13
1.1 Neurčitost v životě	13
1.2 Proč fuzzy databáze?	14
2 FUZZY LOGIKA	15
2.1 Fuzzy množiny	15
2.1.1 Operace s fuzzy množinami	16
2.1.2 Fuzzy čísla	17
2.2 Lingvistické proměnné	19
2.2.1 Sémantika přirozeného jazyka	20
2.3 Evaluační jazykové výrazy	21
2.3.1 Tvary fuzzy množin	21
2.4 Fuzzy databáze	23
2.4.1 Neurčitost v databázích	23
2.4.2 Fuzzy vyhledávání	24
2.4.2.1 Fuzzy SQL	24
3 VÝVOJ INTERNETOVÝCH DYNAMICKÝCH STRÁNEK.....	25
3.1 Programovací jazyky.....	26
3.1.1 HTML	26
3.1.1.1 XHTML	26
3.1.2 Kaskádové styly CSS	26
3.1.3 PHP	27
3.1.4 AJAX	28
3.1.4.1 JavaScript.....	28
3.1.4.2 Document Object Model.....	29
3.1.5 Databázové systémy.....	29
3.1.6 Dotazovací jazyk SQL	29
3.1.6.1 MySQL	30
3.2 Přístupnost webových stránek.....	30

4	EXPERTNÍ SYSTÉM FUZZY NOTEBOOKŮ	31
4.1	Návrh tabulek	31
4.1.1	Tabulka notebooků	31
4.1.2	Tabulka lingvistických proměnných	33
4.1.3	Tabulka fuzzy hodnot	33
4.1.4	Relace	34
4.2	Lingvistické proměnné	35
4.2.1	Výběr parametru notebooků	35
4.2.2	Modelování fuzzy lingvistických proměnných	37
4.2.2.1	Návrh lingvistických proměnných	37
4.2.2.2	Model lingvistických proměnných	39
4.3	Generování fuzzy ohodnocení na základě lingvistických proměnných	39
4.3.1	Výpočet fuzzy ohodnocení	42
4.3.1.1	Automatizovaný výpočet fuzzy ohodnocení	43
4.3.1.2	Kontrola stejných řádků	43
4.4	Vyhledávání notebooků založené na fuzzy expertním systému	44
4.4.1	Pomocné tabulky k vyhledávání	46
4.4.2	Vícekritériální rozhodování	46
4.4.3	Implementace vyhledávání	47
5	E-SHOP S FUZZY EXPERTNÍM SYSTÉMEM.....	49
5.1	Návrh tabulek	49
5.1.1	Tabulka administračního systému	49
5.1.2	Virtuální nákupní košík	50
5.1.3	Tabulka systému k objednání zboží	50
5.1.4	Skladovací systém	51
5.1.5	Tabulka dopravních informací	51
5.1.6	Tabulka diskuse k notebookům	52
5.1.7	Relace	52
5.2	Relace všech tabulek v databázi naší aplikace	52
5.3	Administrační systém	53
5.3.1	Registrace a nastavení účtů	54
5.3.2	Uživatelská práva a jejich přidělování	56
5.4	Katalog notebooků	57
5.4.1	Vložení nového, editace a smazání notebooků	59
5.4.2	Katalog pro uživatele	59

5.4.3	Katalog pro odborníky	60
5.5	Systém k objednání zboží	60
5.5.1	Košík	61
5.5.2	Dokončení objednávky	62
5.5.2.1	Doprava zboží	62
5.5.2.2	Platba za zboží	63
5.5.3	Vyřizování objednávek	64
5.5.3.1	Moje objednávky	64
5.5.3.2	Objednávky k vyřízení	65
5.6	Skladovací systém	66
5.7	Diskusní systém	67
6	ZÁVĚR	69
7	SEZNAM LITERATURY	71
8	SEZNAM OBRÁZKŮ	73
9	PŘÍLOHY	75
9.1	Ukázka stránky aplikace	75
9.2	Model lingvistických proměnných	76
9.3	Vyhledávací dotaz a výsledek jeho vyhledávání	77
9.4	Kompletní relace tabulek v databázi	78

1 ÚVOD

1.1 Neurčitost v životě

Neurčitost se modeluje pomocí matematické disciplíny, která se nazývá fuzzy logika. Díky vývoji skvělých aplikací především v Japonsku a v dalších zemích, získala tato logika velkou popularitu a od počátku devadesátých let se bouřlivě rozvíjí. Byla vytvořena spousta aplikací v oblasti řízení a regulace, k rozpoznávání obrazu, rozhodování, tvorby informačních systémů a pro svou jednoduchost se rychle rozšířila do různých odvětví činnosti člověka. Největší výhodou je, že fuzzy logika umí pracovat s neurčitostí a s výrazy přirozeného jazyka, kterými se člověk dorozumívá při komunikaci s lidmi [1].

Fuzzy logika je založena na vágně definovaném expertním znalostním systému. To je přesný opak toho, co se ve většině případů požaduje, tj. větší přesnost. L. A. Zadeh definoval *princip inkompatibility* (incompatibility principle), který můžeme definovat: „Chceme-li popsat realitu, pak se musíme rozhodnout mezi relevancí informace, která však bude méně přesná, nebo přesností informací, která však bude méně relativní. Při zvyšování přesnosti se dostaneme k bodu, kdy přesnost a relevance se stávají vzájemně se vylučujícími charakteristikami.“ [1]. Příkladem může být funkce internetu. V podstatě k popsání nám stačí několik vágních informací přirozeného jazyka. Pokud však zvýšíme přesnost popisu způsobu funkčnosti internetu, museli bychom začít vysvětlovat od základů, ať se to týká hardware, jeho propojení, až po software využívající zmíněné technologie. Vzniklo by mnoho knih, nad kterými by člověk musel strávit spousty dní, aby to vše přečetl. A i kdyby si udělal čas a přečetl by je, i tak by bylo nutné k pochopení použít určitý stupeň vágnosti, tj. vrátili bychom se k vágní charakterizaci, jinak by se člověk v textu ztratil [1].

V předchozím textu jsme se zmínili o dvou důležitých termínech. Relativní informaci popíšeme přirozeným jazykem, který je velmi efektivním nástrojem a dosud jediný možný pro práci s neurčitými výrazy. U přesnosti lze říci, že jde jen o iluzi, protože jí nelze dosáhnout. Kdyby jí bylo možné dosáhnout, znamenalo by to, že při různých přesnostech bychom získali neustále stejné hodnoty. Např. uvažujme absolutní délku dvou hranolů, což by znamenalo, že oba hranoly jsou stále stejně velké, ať měříme v mm, μm , pm, nm, apod. To nás dovede k samotným atomům nebo k elementárním částicím, u kterých si můžeme položit otázku: Jsme schopni rozeznat, který atom či elementární částice je element hranolu a který ne? Vždy nás totiž dovede ke sporu úsilí o absolutní přesnost. Není nutné ztrácet naději, protože náš přirozený jazyk umí dokonale využít vágnost, která je jeho největší silou [1].

Náš přirozený jazyk neslouží pouze k dorozumění mezi lidmi, ale je potřeba si uvědomit, že je významnou součástí u různých lidských aktivit, které lze popsat. Např. při učení plavání nám instruktor vysvětlí techniku plavání v přirozeném jazyce. Např. musíte pomalu kopat nohama a zároveň kroužit rukama podél těla, apod. Instruktor svůj výklad navíc doplňuje názornými ukázkami. Takto vysvětlené techniky plavání nám dostačují k tomu, abychom se naučili plavat, a přitom jde o velmi vágní pokyny [1].

Všechny výše definované poznatky mají souvislost s úvahami zakladatelů fuzzy logiky, mezi které především patří profesor kalifornské univerzity v Berkeley Lotfi A. Zadeh. Fuzzy logika se zabývá vágností, kterou se pokoušíme matematicky popsat a jejím základem je teorie fuzzy množin. Můžeme si položit otázku, jestli fuzzy logika není

podobná pravděpodobnosti a statistice. Pro zodpovězení je potřeba si uvědomit, že pravděpodobnost se zabývá modelováním nejistoty, která buď nastane, nebo nenastane. Pomocí pravděpodobnosti zjistíme, že jeden z jevů nastane s větší pravděpodobností než druhý. „Pokud jsou jevy nezávislé, pak pravděpodobnost toho, že nastanou současně, je rovna součinu jejich pravděpodobností.“ [1].

Uvažujme např. jev „všechny měděné kuličky“. Nejdříve musíme definovat, co je to měděný a co je to kulička. U fuzzy logiky se nejedná o to, zda daný stav nastane či nikoliv. Základem fuzzy množin jsou stupně příslušnosti, které jsou v rozmezí [0 až 1]. „S trochou nadsázky lze tedy říci, že pravděpodobnost nám odpovídá na otázku, zda něco nastane, zatímco teorie fuzzy množin nám odpovídá, co nastalo.“ [1]. V realitě se vždy setkáme s nejistotou a vágností. V této práci se budeme zabývat jen teorií fuzzy množin, neboť teorie, která by řešila oba způsoby, zatím neexistuje [1].

Nabízí se otázka, jestli by nebylo možné nějakým způsobem nejistotu a vágnost odstranit? Nejistota je v přírodě silně ukotvena a nelze přesně definovat, co nastane a co nikoliv. Jak již bylo řečeno výše, vágnost nelze chápat jako chybu, kterou odstraníme přesnějším měřením. Vágnost je součástí přirozeného jazyka, kterým se domlouváme. Např. představme si, že by nám někdo při vysvětlování cesty řekl, půjdete 257,678 m, tam se pootočíte o 19°33'25" doleva a jste na místě. Na základě těchto informací bychom se na dané místo dorazili jen s velkou námahou, i kdybychom využili nejpřesnějších měřicích zařízení, a přitom nám stačí vágní informace, např. půjdete asi 3 minuty a pak mírně zahnete doleva a jste na místě [1].

1.2 Proč fuzzy databáze?

Mezi aplikace, které využívají fuzzy logiku, se řadí různé domácí spotřebiče, především automatické pračky, které dokáží poznat znečištění prádla, na základě kterého volí program praní, stejně tak vysavač na základě druhu podlahy a velikosti nečistot volí sílu sání. „Dále je fuzzy logika využívána při automatizovaném řízení podzemní dráhy ve městě Sendai v Japonsku“ [1]. V automobilovém průmyslu, např. „automobilka Nissan jako jedna z prvních zavedla na počátku devadesátých let fuzzy logiku v ABS protiskluzovém systému a automatické převodovce. Automobilka Volkswagen nabízí ABS automatickou převodovku, která reaguje na styl jízdy řidiče (sportovní, opatrný, apod.) a podle toho řadí rychlosti.“ [1]. Fuzzy logika u vozidel vybavených klimatizací řídí otáčky volnoběhu. Dalšími příklady využití je rozpoznávání ručně napsaného písma, obchodování s akcemi ve finančním sektoru, hodnocení firem při schvalování úvěrů, atd.

Fuzzy logika se používá i v databázích, které většinou obsahují přesná data o dané problematice. Protože člověk lépe pochopí informace v přirozeném jazyce, i na úkor přesnosti můžeme databázový systém s přesně definovanými informacemi vybavit fuzzy expertním systémem, na základě kterého můžeme s databází komunikovat v přirozeném jazyce a využívat vágního vyhledávání, jako např. „lidé ve středním věku s malým platem.“ Další možnost využití je při nakupování v internetovém obchodě v případě, že tento internetový obchod bude vybaven expertním znalostním systémem, který bude rozumět aktuálním trendům a technickým parametrům zboží. V případě, že zákazník nemá dostatečné informace o trendech poslední doby, můžeme mu v internetové aplikaci formou vyjádření v přirozeném jazyce ulehčit nákup. Takový internetový obchod bude schopný poradit podobně jako obsluha v klasických obchodech. Jak internetový obchod s expertním znalostním systémem navrhnout se dozvíme v dalším textu. V další kapitole zavedeme základní pojmy z teorie fuzzy logiky, které budeme dále využívat.

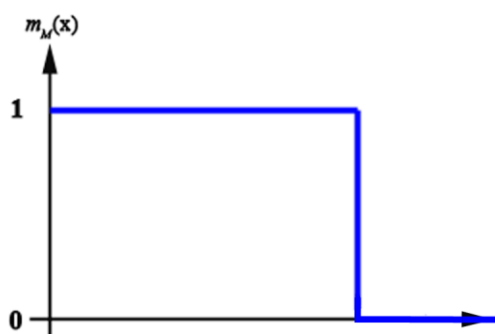
2 FUZZY LOGIKA

2.1 Fuzzy množiny

Fuzzy množiny jsou základem fuzzy logiky, obsahují neurčité údaje a jejich míry ohodnocení. Základní množiny lze popsat několika způsoby, mezi které především patří [2]:

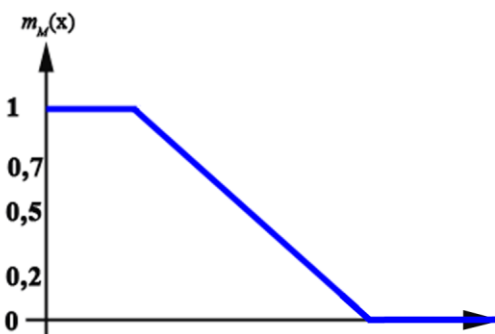
- Definujme množinu prvků $M = \{x_1, x_2, \dots, x_n\}$.
- Množinu lze popsat charakteristickou funkcí $m_M(x)$, pro kterou platí:

$$m_M(x) = \begin{cases} 1, & \text{jestliže prvek } x \text{ do množiny patří} \\ 0, & \text{jestliže } x \text{ do množiny nepatří} \end{cases}$$



Obr. 1 Skoková funkce klasické množiny [2].

U klasických množin hovoříme o tzv. *ostrých* (*crisp*) množinách. Prvek x do dané množiny patří nebo nepatří. Pokud k dané množině doplníme stupeň příslušnosti, tj. ke každému prvku v množině přiřadíme hodnotu od 0 do 1, pak to ve fuzzy logice znamená, s jakým stupněm příslušnosti daný prvek do množiny patří nebo nepatří a tím dosáhneme neostré funkce [2].



Obr. 2 Průběh funkce fuzzy množiny [2].

Fuzzy množiny vychází z následující myšlenky. Chceme-li např. definovat množinu všech mladých lidí, pak je nejprve potřeba definovat, co je to mladý člověk. Řekněme, že jde o lidi ve věku od 16 do 30 let, tj. $U = [16, 30]$. Problém ale nastane, když má člověk např. 30 let a 2 měsíce. Pak podle předchozí definice by to tedy znamenalo, že již není mladý, i když v podstatě tyto dva věky nelze rozeznat. Proto ve fuzzy logice

je vhodné použít celou množinu stárí lidí, tj. $U = [0, 120]$ a pro každý věk definovat stupeň příslušnosti pro danou množinu. Stupeň číslo 0 znamená absolutní nepravdu a hodnota 1 znamená absolutní pravdu. Hodnoty mezi 0 až 1 určují částečnou pravdu, která je tím vyšší, čím je vyšší stupeň příslušnosti. V našem případě množiny mladých lidí, např. člověk, který má věk 12 let, do množiny mladých lidí patří s ohodnocením 0. Člověk, který by měl 23 let, do množiny mladých lidí patří s ohodnocením 0,5 a člověk, který by měl 29 let, patří do množiny mladých lidí s ohodnocením 1 [15].

Obdobné úvahy můžeme provést pro různá slova přirozeného jazyka, např. velikost, výška, rostliny, auta, notebooky, atd. Zvolené slovo musí být nějakým způsobem změřitelné nebo porovnatelné. Při návrhu fuzzy množiny tedy nejdříve navrhne U nazvané univerzum, které může být libovolnou množinou, jak již bylo řečeno výše. Matematicky lze fuzzy množinu s využitím pojmu univerzum definovat tímto předpisem: $A: U \rightarrow [0,1]$.

To znamená, že fuzzy množina je reprezentována vhodnými prvky x z množiny U a ke každému prvku je přiřazeno přirozené číslo v rozmezí od 0 do 1, které nazýváme *stupněm příslušnosti* prvku x do fuzzy množiny A [16].

2.1.1 Operace s fuzzy množinami

Fuzzy množiny umožňují provádět všechny operace jako u klasických množin. Základními operacemi jsou: sjednocení, průnik a doplněk. Kromě základních operací však fuzzy množiny nabízí další operace, které v klasické teorii jsou ekvivalentní se základními operacemi nebo nemají smysl [14].

Základní operace s fuzzy množinami

Sjednocení (logický součet): $C = A \text{ or } B$ nebo $C = A \cup B$

Sjednocení dvou fuzzy množin A a B je množina C , kde stupně příslušnosti jejich prvků je definovaná následujícím předpisem: $m_C = m_{A \cup B(x)} = \max\{m_{A(x)}, m_{B(x)}\}$ [2]

Příklad: Máme množinu $A = [1/0,4; 2/0,2; 3/0,8]$ a $B = [1/0,9; 2/0; 3/0,5]$, z kterých vznikne množina $C = [1/0,9; 2/0,2; 3/0,8]$

Průnik (logický součin): $C = A \text{ and } B$ nebo $C = A \cap B$

Průnik dvou fuzzy množin A a B je množina C se stupni příslušnosti prvků definovaných takto: $m_C = m_{A \cap B(x)} = \min\{m_{A(x)}, m_{B(x)}\}$ [2]

Příklad: Máme množinu $A = [1/0,4; 2/0,2; 3/0,8]$ a $B = [1/0,9; 2/0; 3/0,5]$, z kterých vznikne množina $C = [1/0,4; 2/0; 3/0,5]$

Doplněk

Doplněk fuzzy množiny A je množina C se stupni příslušnosti prvků definovaných takto: $m_{C(x)} = 1 - m_{A(x)}$ [2]

Příklad: Máme množinu $A = [1/0,4; 2/0,2; 3/0,8]$, z které vznikne množina $C = [1/0,6; 2/0,8; 3/0,2]$.

Doplněk není totéž co negace. Např. „ne mladí lidé“ znamená, že jde o množinu nemladých lidí. Obecně však nejsme schopni říci, co znamená být „mladý“, proto však může nastat situace, že daný věk člověka patří do množiny mladý a současně i do množiny ne mladý [1].

Další operace s fuzzy množinami

Tyto operace popíšeme jen stručně pro zajímavost, protože v dalším textu již nejsou použity. Pro podrobnější informace o daných operacích doporučujeme odbornou literaturu [1].

Ve fuzzy logice se používají dvě operace sjednocení a průniku. V případě potřeby přesnějšího sjednocení a průniku použijeme Lukasiewiczovu konjunkci (sjednocení) a disjunkci (průnik) [1].

Lukasiewiczova konjunkce (sjednocení): $a \otimes b = 0 \vee (a + b - 1)$

Lukasiewiczova disjunkce (průnik): $a \oplus b = 1 \wedge (a + b)$

T-normy a zobecnění operací s fuzzy množinami

Binární funkce $T: [0,1] \times [0,1] \rightarrow [0,1]$ je takzvaná t-norma, která vyhovuje následujícím vlastnostem pro všechna $x, y, z \in [0,1]$

- a) komutativnost: $T(x, y) = T(y, z)$
- b) asociativnost: $T(x, T(y, z)) = T(T(x, y), z)$
- c) monotónnost: Jestliže $x \leq y$, pak $T(x, z) \leq T(y, z)$
- d) omezenost: $T(0, x) = 0$ a $T(1, x) = x$

Základní t-normy jsou:

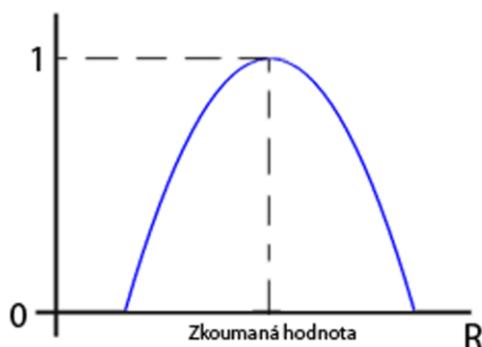
- a) operace minima: $M(x, y) = x \wedge y = \min(x, y)$
- b) $P(x, y) = x \cdot y$
- c) Lukasiewiczova konjunkce: $T_{\infty}(x, y) = 0 \vee (x + y - 1)$
- d) tzv. drastický součin, tj. nejmenší t-norma: $W(x, y) = \begin{cases} x \wedge y, \text{ jestliže } x \vee y = 1 \\ 0, \text{ jinak} \end{cases}$

Jestliže je definovaná t-norma, pak jsou definovány i odpovídající t-konormy [1]

- a) operace maxima: $M(x, y) = x \vee y = \max(x, y)$
- b) $Q(x, y) = x + y - x \cdot y$
- c) Lukasiewiczova disjunkce: $S_{\infty}(x, y) = 1 \wedge (x + y)$
- d) tzv. drastický součin, tj. nejmenší t-konormy: $V(x, y) = \begin{cases} x \vee y, \text{ jestliže } x \wedge y = 0 \\ 1, \text{ jinak} \end{cases}$

2.1.2 Fuzzy čísla

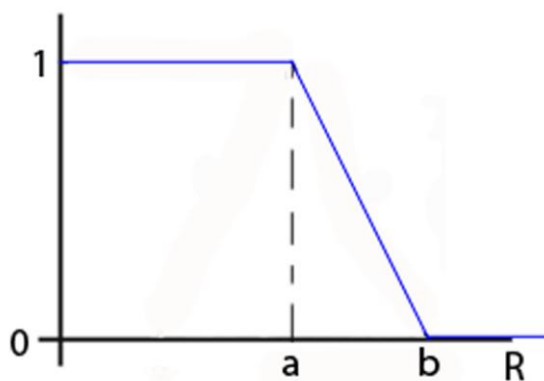
Fuzzy čísla jsou vágně definované hodnoty, např. „asi 10“, „asi 21°C“. Při měření metrem lze zjistit, že měřená místnost má na délku 5,5 m, ale protože jsme si v úvodu tohoto textu řekli, že nejsme v podstatě schopni přesného měření, vždy dosáhneme vágních hodnot, protože pokud bychom použili přesnější měřidlo, zjistíme, že místnost má 5,47 m, ale přitom přesná hodnota je 5,455 m. Naše měření je vždy nepřesné a to i přes to, že použijeme nejpřesnější typ měřidel, který máme k dispozici, např. pokud měříme v metrech, automaticky nám vzniká nepřesnost v dm, cm, mm, atd. Fuzzy čísla lze definovat jako speciální typ fuzzy množin reálných čísel $\mathbb{R} = (-\infty, +\infty)$, znázornit jej můžeme pomocí Obr. 3 [1].



Obr. 3 Funkce příslušnosti fuzzy čísla [1].

Základní funkce a tvary fuzzy čísel:

Minimální funkce

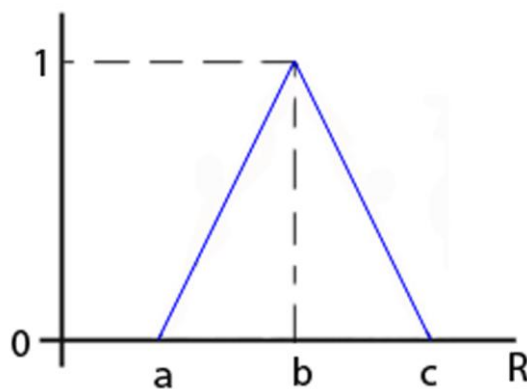


Obr. 4 Minimální funkce [1].

Pro získání fuzzy čísel ve všech funkčních oblastech minimální funkce lze definovat následující matematický popis: [1]

$$Z(a,b) = \begin{cases} 1, & x \leq a \\ \frac{b-x}{x-a}, & a \geq x \geq b \\ 0, & x \geq b \end{cases}$$

Střední funkce

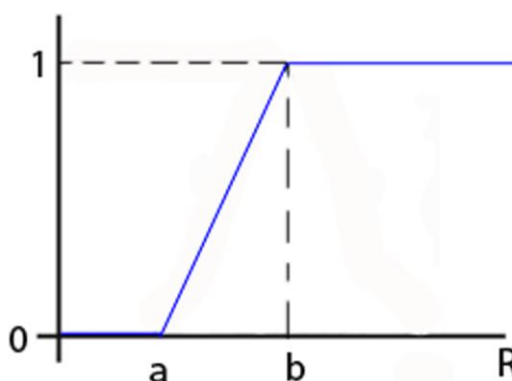


Obr. 5 Střední funkce [1].

Pro získání fuzzy čísel ve všech funkčních oblastech střední funkce lze definovat následující matematický popis:

$$Z(a,b,c) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \\ 1, & x \geq c \end{cases}$$

Maximální funkce



Obr. 6 Maximální funkce [1].

Pro získání fuzzy čísel ve všech funkčních oblastech maximální funkce lze definovat následující matematický popis:

$$Z(a,b) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & x \geq b \end{cases}$$

Podrobnější popis bude uveden v kapitole 2.2.3. Výše uvedené fuzzy funkce úzce souvisí s lingvistickými proměnnými, jež jsou definované přirozeným jazykem, kterým se věnuje následující kapitola [1].

2.2 Lingvistické proměnné

Hodnoty výrazů přirozeného jazyka se nazývají *lingvistické proměnné*. „Hodnotu lingvistické proměnné můžeme definovat jako neostrou fuzzy množinu.“ [2]. Tato množina se nazývá množina termů, které jsou vymezeny na univerzu a posuzujeme je jako univerzální množinu [1].

Např. „teplota okolí“ je hodnota lingvistické proměnné. Ve většině případů se teplota okolí měří ve stupních Celsia, ale při slovním vyjádření většinou mluvíme o teplotě jako např. „je zima“, „teplo“, „horko“, atd. Pro předchozí neostré lingvistické proměnné můžeme definovat funkce příslušnosti [1].



Obr. 7 Funkce příslušnosti [2].

Funkci příslušnosti neostré množiny vysvětlíme na Obr. 7, z kterého je vidět, že máme definovány dvě lingvistické proměnné a v univerzu je funkční rozmezí teplot pro dané lingvistické proměnné.

- Pokud je daná hodnota 100°C nebo méně než 100°C , pak jde o lingvistickou proměnnou „HORKÁ VODA“ z funkcí příslušnosti 1.
- Pokud je daná hodnota 45°C , pak jde o lingvistickou proměnnou „HORKÁ VODA“ se stupněm ohodnocení 0,5.
- Pokud je daná hodnota 25°C , pak jde o lingvistickou proměnnou „HORKÁ VODA“ z funkcí příslušnosti 0,15 a zároveň lingvistická proměnná „VLAŽNÁ VODA“ má funkci příslušnosti 1.
- Pokud je daná hodnota 15°C , pak jde o lingvistickou proměnnou „VLAŽNÁ VODA“ s funkcí příslušnosti 0,5 a zároveň lingvistická proměnná „HORKÁ VODA“ má funkci příslušnosti 0.
- Pokud naměříme 7°C a více, pak obě lingvistické proměnné mají funkci příslušnosti 0 [2].

2.2.1 Sémantika přirozeného jazyka

Jak jsme si výše řekli, hlavní silou fuzzy logiky je možnost využití významu slov a některých jednoduchých výrazů přirozeného jazyka. Použití sémantiky přirozeného jazyka tak, aby se použilo částí nebo celých vět přirozeného jazyka, je velkým problémem. Museli bychom nejprve vystihnout sémantiku sloves, což je nejtěžší prvek vět. Z těchto důvodů L. A. Zadeh zavedl tzv. *jazykovou proměnnou*, což jsou slova nebo části přirozeného jazyka, např. „mladý nebo středně mladý“, apod. [1].

Při modelování sémantiky přirozeného jazyka je důležité pochopit trojici pojmů: intenze, možný svět a extenze.

- **Intenze:** jde o reprezentaci vlastnosti, která je v čase i místě neměnná a popisuje daný výraz, např. pojem „nízká teplota“ - její intenzí je „být nízký“. O samotných objektech nemá smysl hovořit, dokud nejsou konkrétně definovány. Může jít např. o teplotu vody, vzduchu, těla, apod.
- **Možný svět:** intuitivně lze říci, že jde o popis všech možných (myslitelných) věcí kolem nás. Teprve v možném světě můžeme hovořit o konkrétních objektech.

- **Extenze:** Množina všech objektů definovaná intenzí v možném světě se nazývá extenze. Z následující definice plyne, že jedna intenze odkazuje na množinu extenzí skutečných objektů v možném světě, např. možný svět je teplota vzduchu v okolí - její intenze je „vysoká teplota“ a jejich extenze je množina teplot okolo 30°C [1].

Pojem extenze a intenze je potřeba od sebe odlišovat, protože definovaná vysoká teplota může být fuzzy hodnotou pro určitý možný svět, např. pro střední Evropu. Kdybychom chtěli určit množinu vysokých teplot, např. v jihovýchodní Africe, musela by extenze být v rozmezí okolo 55°C apod. [1].

2.3 Evaluační jazykové výrazy

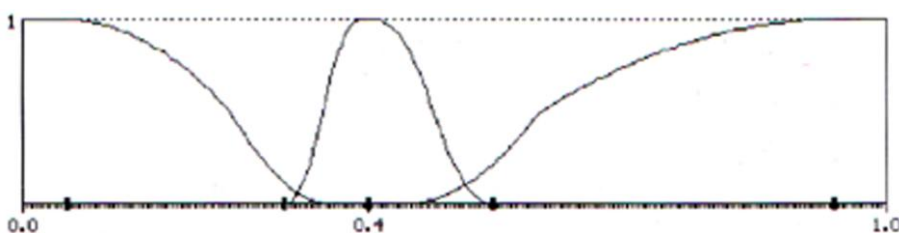
Pro modelování fuzzy množin jsou důležité evaluační jazykové výrazy, které charakterizují většinou nějaká čísla pomocí přirozeného jazyka. Jde o hodnoty, které lze bez problémů popsat slovně, např. výrazy definující:

- **místo:** „vlevo, více vlevo, ve středu, vpravo, více vpravo ...“,
- **velikost:** „nejmenší, malý, větší, největší ...“,
- **kvalita výrobku:** „nejkvalitnější, kvalitní, neutrální, nekvalitní ...“,
- **popis vzhledu:** „nejkrásnější, krásná, hezká, normální, nehezká, škaredá ...“.

Výše uvedené evaluační jazykové výrazy lze i propojovat prostřednictvím spojek „a (and)“, anebo „nebo (or)“. Jde o požadavky, které jsou kladeny na fuzzy model. Někdy pomocí spojek můžeme zjednodušit popis modelu [1].

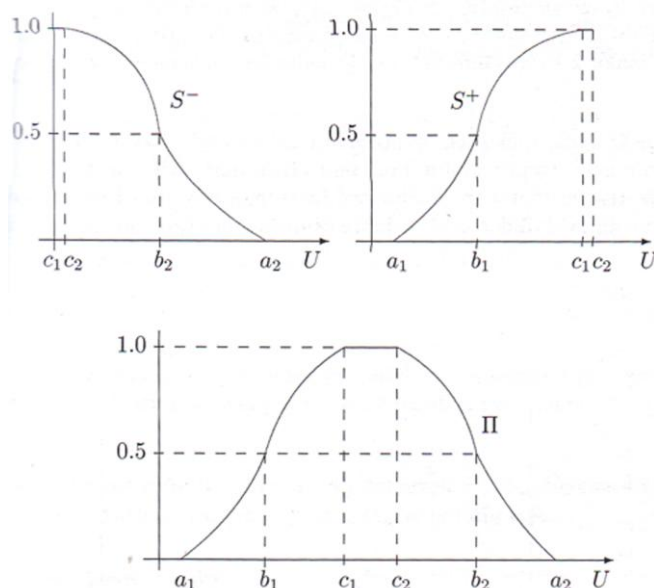
2.3.1 Tvary fuzzy množin

Protože tvary fuzzy množin jsme již definovali v kapitole 2.1.2, v následující kapitole doplníme informace o možných tvarech fuzzy množin. Jak už víme, základem modelování sémantiky přirozeného jazyka jsou fuzzy množiny. Obecně se akceptuje, že vybrané množiny odpovídají evaluačním výrazům „malý, střední, velký, ...“ (tzv. základní evaluační trichotomie) má tvar [1]:



Obr. 8 Základní evaluační trichotomie [1].

Fuzzy množiny mohou mít kromě základních tvarů definovaných v kapitole 2.1.2 následující tvary:

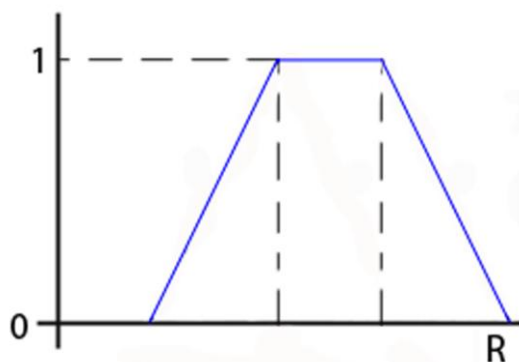


Obr. 9 Tvary fuzzy množin [1].

K výše uvedeným fuzzy tvarům můžeme určit funkci příslušnosti na základě následující rovnice [1]:

$$F(x, a_1, a_2, b_1, b_2, c_1, c_2) = \begin{cases} 0, & x \leq a_1 \text{ nebo } x \geq a_2 \\ \frac{1}{2} \left(\frac{x - a_1}{b_1 - a_1} \right)^2, & a_1 < x < b_1 \\ 1 - \frac{1}{2} \left(\frac{c_1 - x}{c_1 - b_1} \right)^2, & b_1 < x < c_1 \\ 1 - \frac{1}{2} \left(\frac{x - c_2}{b_2 - c_2} \right)^2, & a_1 < x < b_1 \\ \frac{1}{2} \left(\frac{a_2 - x}{a_2 - b_2} \right)^2, & b_2 < x < a_2 \\ 1, & c_1 \leq x \leq c_2 \end{cases}$$

V některých případech se nám může např. obdélníkový tvar hodit více než trojúhelníkový tvar (viz. Obr. 5), uvedený v kapitole 2.1.2.



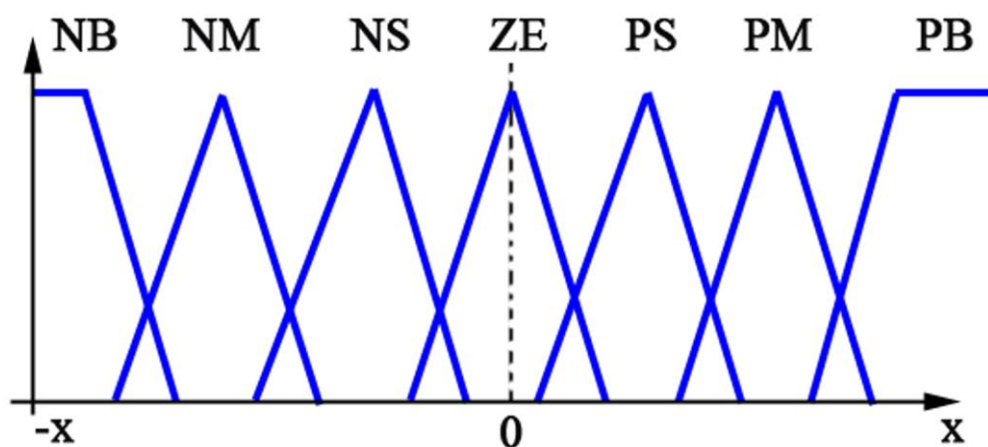
Obr. 10 Obdélníkový tvar [1].

Ve většině případů se používá zjednodušených trojúhelníkových tvarů. V některých případech, zejména v regulaci, je vhodné využít i zjednodušených jazykových proměnných.

Tab. 1 Zjednodušené jazykové proměnné [2].

Význam	Ozn. Čes.	Ozn. Ang.
Hodnota velká záporná	ZV	NL
Hodnota střední záporná	ZS	NM
Hodnota malá záporná	ZM	NS
Hodnota záporná blízká nule	ZN	NZ
Hodnota nulová	NU	Z
Hodnota kladná blízká nule	KN	PZ
Hodnota malá kladná	KM	PS
Hodnota střední kladná	KS	PM
Hodnota velká kladná	KV	PL

Výsledný tvar fuzzy množin vytvořený ze zjednodušených jazykových proměnných je na obr. 11. Podle svého tvaru se nazývá „pila“ [1].



Obr. 11 Tvary zjednodušených jazykových proměnných [1].

2.4 Fuzzy databáze

Databáze se používají k ukládání dat do relačních databází. Data mohou být různého druhu - od osobních údajů přes databáze zboží až po specifická data různých speciálních systémů. V databázích jsou uložena určitá data popisující nějaký objekt [6].

2.4.1 Neurčitost v databázích

Pokud požadujeme do databáze ukládat i neurčitá data, pak musíme vedle klasických dat vytvořit fuzzy logiku, která nám bude danou databází obsluhovat. Databáze nám poslouží jako zdroj dat fuzzy logiky, která bude z databáze čerpat lingvistické proměnné i jejich stupně příslušnosti.

Otázkou je, k čemu je dobrá nepřesnost v databázích. Odpověď je zcela jednoduchá. Představme si, že máme v databázi prospěch třídy a požadujeme najít žáky, kteří mají výborný prospěch. Můžeme určit, že výborný prospěch je známka lepší nebo rovna aritmetickému průměru 2. Až potom můžeme použít klasickou databázi. Ale co když máme v databázi žáka, který má prospěch 2,03. V klasické databázi by to znamenalo, že tento žák by už nebyl v seznamu výborných žáků, přestože jde jen o velmi malou odchylku od prospěchu 2. K tomuto účelu je výhodné použít fuzzy databáze a ukládat do databáze neurčité údaje, tj. kromě konkrétních čísel i jejich lingvistické proměnné a jejich stupně ohodnocení, a pak není nic jednoduššího, než si v databázi vyhledat všechny žáky s výborným prospěchem, aniž bychom museli řešit, co to znamená výborný prospěch, protože tato informace je již definována při modelování fuzzy logiky.

2.4.2 Fuzzy vyhledávání

Největší silou fuzzy logiky implementované v databázích je možnost vágního vyhledávání kdy nám tato logika umožňuje komunikovat s databází, tak jak jsme zvyklí v přirozeném jazyce nebo přesněji řečeno pomocí evaluačních výrazů jazyka. Předpokládejme, že máme k dispozici databázi žáků. Můžeme si vyhledat žáky s výborným prospěchem a zároveň malým počtem zameškaných hodin. Máme-li k dispozici účetní databázi nějaké firmy, s využitím fuzzy logiky můžeme jednoduše zjistit seznam všech mladých lidí, kteří mají vysoký příjem.

2.4.2.1 Fuzzy SQL

SQL (Structured Query Language) je strukturovaný dotazovací jazyk sloužící k práci s relačními databázemi a jejich daty [6].

Dotazovacím jazykem SQL se budeme podrobněji zabývat v kapitole 3.2.1. Nyní uvedeme jen způsob použití SQL pro fuzzy databáze. Zásadou je, nepoužívat u fuzzy databází při vyhledávání žádné přesné hodnoty, ale využívat lingvistické proměnné. Např. řekněme, že mladý člověk je každý, kdo má 27 nebo méně let. Pokud budeme chtít zjistit seznam všech mladých lidí, tak v klasickém SQL dotazu bychom napsali:

```
SELECT * FROM ZAMESTNANEC WHERE VEK <= 27
```

V případě použití fuzzy SQL dotazu:

```
SELECT * FROM ZAMESTNANEC WHERE LINVEK = 'MLADY'.
```

Stejně bychom mohli pokračovat, např. když bychom chtěli zjistit všechny zaměstnance s vysokým příjmem. Řekněme, že vysoký příjem je vyšší jak 30 000,- Kč.

Klasický SQL dotaz by vypadal následovně:

```
SELECT * FROM ZAMESTNANEC WHERE PRIJEM >= 30000
```

V případě použití fuzzy dotazu:

```
SELECT * FROM ZAMESTNANEC WHERE PRIJEM  
= 'VYSOKY_PRIJEM'
```


3 VÝVOJ INTERNETOVÝCH DYNAMICKÝCH STRÁNEK

Internet, přesněji řečeno *služba www* (World Wide Web), která v roce 2009 slaví své dvacáté narozeniny, se za dobu své existence stal součástí života více než třetiny lidí žijících na planetě. Internet byl vytvořen v období studené války, kdy byl požadavek vytvořit síť, která by fungovala i po vyřazení některé její části. Začal vývoj protokolu *TCP/IP*, který je jedním z nejdůležitějších protokolů pro komunikaci počítačů po sítích a tím pádem i na internetu. Zárodkem internetu byla experimentální síť *Arpanet* spuštěná v roce 1969 a vypnutá v roce 1990. Byla navržena služba *www*, která měla původně sloužit pro rychlé a pohodlné zveřejňování vědeckých článků badatelů v oblasti vysokoenergetických fyzikálních polí, mezi geograficky vzdáleně rozloženými místy. Tim Berners-Lee, který tenkrát pracoval jako konzultant v organizaci Cern zabývající se jaderným výzkumem – sídlící v Ženevě, kde na konci 80. let navrhl hypertextový systém a k přenosu dat použil *protokol HTTP* (Hyper Text Transfer Protocol). Stejný autor vytvořil značkovací jazyk *HTML* pro vývoj *www* stránek a dále navrhl první *webový prohlížeč*, který nazval *WorldWideWeb*. V roce 1989 poprvé publikoval návrh vývoje služby *www* a v roce 1990 spustil první webový server s *www* stránkou, který běží dodnes na internetové adrese <http://info.cern.ch>. V roce 1992 přestal být internet využíván jen pro vědecké a vojenské účely jako doposud a služba *www* byla zpřístupněna i pro komerční účely a tedy i veřejnost. O internet byl velký zájem. Již rok od spuštění fungovalo 60 serverů, na kterých běžela služba *www*. V roce 1993 byl představen první *grafický prohlížeč* nazvaný *Mosaic*, který navrhli studenti NCSA (National Center of Supercomputer Applications). S rychlým rozšiřováním internetu a oblibou mezi lidmi bylo nutno vytvořit jednotnou adresu pro všechny služby, které internet nabízel. K tomuto účelu se navrhla adresa s názvem *URL* (Uniform Resource Locator) a její tvar je:

typ://uživatel:heslo@počítač:port/cesta;parametry?dotaz [9]

- ✓ **TYP** - název služby např. *http*, *ftp* atd.,
- ✓ **UŽIVATEL a HESLO** - přihlašovací údaje uživatele,
- ✓ **POČÍTAČ** - identifikace počítače (jméno počítače nebo IP adresa),
- ✓ **PORT** - číslo komunikačního portu *TCP/IP*,
- ✓ **CESTA** - označení požadovaného souboru včetně cesty k němu,
- ✓ **PARAMETRY a DOTAZ** - blíže specifikují požadavky na server a jejich význam se v různých aplikacích liší [7].

Služba *www* je typu klient/server, kde klientem je prohlížeč a server je místo, kde je uložena internetová stránka, ke které se připojujeme pomocí *URL* adresy. Jelikož IP adresy se lidem hůře pamatují, byla navržena textová jména počítačů, která jsou ve většině případů intuitivní a uživatelům se tím pádem lépe pamatují. K tomu účelu byla navržena *služba DNS* (Domain Name System). *DNS* se skládá z kořenové domény, která není pojmenována. Pod kořenovou doménou je geografické členění podle států (*cz*, *sk*, *fm*, *ch*, apod.) kromě např. *USA* nebo podle typu instituce (*com*, *net*, *info*, *org*, apod.), např. v *USA*. Pod tímto rozdělením není další větvení omezeno. Jediným nejzásadnějším požadavkem je, aby jména v každé doméně byla unikátní [9].

V dnešní době, kdy se internet stále více zrychluje, dochází k rychlému rozvoji *www* aplikací, kde na internetu najdeme dynamické aplikace, které reagují na požadavky

uživatelé, ať mluvíme o informačních systémech, e-shopech, redakčních systémech nebo spouštění internetových aplikací, které původně byly desktopovými (např. kancelářský balík od Google, Google Earth nebo aplikace pro úpravu obrázků od společnosti Adobe). Do budoucna se předpokládá, že by takových aplikací mohlo být na internetu mnohem více. Dokonce se předpokládá, že v budoucnu by mohl být operační systém načten přímo z nějakého serveru, který je připojený k internetu (tzv. cloud computing) [10].

3.1 Programovací jazyky

K vývoji internetových aplikací se používají různé programovací jazyky. Na počátku internetu byly www stránky naprogramovány v značkovacím jazyku HTML. V dnešní době se však používají k vývoji www aplikací různé programovací jazyky, např. php, java, ASP NET, atd. V tomto textu popíšeme jen jazyky, které byly použity při vývoji aplikace.

3.1.1 HTML

HTML (*Hypertext Markup Language*) je značkovací jazyk, který slouží k vytvoření statických stránek pomocí hypertextových odkazů pro www služby. Vznikl jako podmnožina z dříve vyvinutého univerzálního značkovacího jazyka SGML (*Standard Generalized Markup Language*). HTML je standardizován od roku 2000 jako ISO norma 15445 [10].

V roce 1989 se k vytváření textových dokumentů používal TeX, PostScript a také SGML. Tim Berners-Lee si uvědomil, že je nezbytné k publikování článků na www stránkách využít něco jednoduššího a tak v roce 1990 byl navržen jazyk HTML. Tvůrci původně předpokládali, že se vytvoří aplikace pro generování HTML kódu založená na WYSIWYG principech [9]. V dnešní době existuje takových generátorů mnoho, protože však jejich výstup není moc kvalitní, z těchto důvodů programátoři píšou www stránky přímo v jazyce HTML. Vývoj jazyka má na starosti W3C, protože definice jazyka byla ovlivňována nejrozličnějším rozšířením autorů webových prohlížečů, nebyla jeho úloha v procesu sjednocování jednoduchá [10].

3.1.1.1 XHTML

XHTML (*Extensible Hypertext Markup Language*) je rozšířený značkovací jazyk HTML. Je vyvinutý také v konsorciu W3C. Předpokládalo se, že do budoucna XHTML nahradí HTML, u kterého byl vývoj ve verzi 4.01 ukončen. V roce 2007 byl však započat vývoj HTML 5 a její XML varianty XHTML 5. Vedle toho však i paralelně běží vývoj XHTML 2.0. **Základní rozdíly XHTML oproti HTML**

- V XHTML je nutno, aby byly všechny tagy ukončené, a to i nepárové. Ukončení se provádí ``,
- v XHTML je nutno, aby všechny elementy byly zapsány malými písmeny,
- všechny hodnoty atributů musí být uzavřeny do uvozovek,
- dokument musí začínat **XML deklarací** [10].

3.1.2 Kaskádové styly CSS

CSS (*Cascading Style Sheets*) jsou předpisy kaskádových stylů. Jde především o jazyk, který slouží k popisu způsobu zobrazení stránek napsaných v HTML, XHTML nebo XML. Jazyk opět navrhla standardizační konsorcia W3C, první návrh kaskádových stylů navrhl Håkon Wium Li [10].

Hlavní myšlenkou CSS je elegantní možnost oddělení vzhledu a především grafiky www stránky od režijní struktury a obsahu stránky [9]. Při vývoji www služby se předpokládalo, že vzhled stránky bude implementován přímo v značkovacím jazyce HTML, ale z důvodů nedostatků standardů v počátcích vývoje a konkurenčního boje výrobců webových prohlížečů si vynutil jiný způsob řešení [10].

Výhody [10]

- Lepší formátovací schopnost než v HTML,
- zajištění stejného vzhledu nadpisů, seznamů, odstavců, tabulek, atd.,
- změna vzhledu se provede přepsáním části nebo celého css souboru,
- formátování XML dokumentů,
- větší kompatibilita alternativních webových prohlížečů,
- kratší doba načítání stránky.

Nevýhody [10]

- Zásadní nevýhodou, která ztěžuje práci všem webovým designérům, je ne vždy dobrá podpora od nejrozšířenějších výrobců webových prohlížečů, které obsahují chyby CSS a v některých případech není úplně jednoduché dosáhnout stejného vzhledu ve všech nejčastěji veřejností využívaných prohlížečích. Situace se v posledních letech značně zlepšuje s nástupem nových verzí, především Internet Exploreru [10].

3.1.3 PHP

PHP zkratka znamená: *Hypertext Preprocessor* - hypertextový preprocesor, původně znamenala *Personal Home Page* [11]. Je to skriptovací programovací jazyk, který především slouží k programování dynamických internetových stránek. Ve většině případů se PHP vkládá přímo do struktury jazyka HTML nebo XHTML. PHP skript běží na straně serveru. Internetový prohlížeč požádá server o určitou internetovou stránku. V případě, že server při zpracovávání stránky narazí na php skript, předá tento skript php překladači, který vygeneruje výstup do HTML stránky nebo jiného výstupu, a takto nějakým způsobem naformátovaný výstup předá server stránku prohlížeči, který jí zobrazí uživateli [3].

O vývoj PHP se zasloužil Rasmus Lerdorf v roce 1994, kdy napsal binární část Common Gateway Interface (CGI) v programovacím jazyku C. PHP bylo vytvořeno pro editaci osobních domácích stránek, pro záměnu s malou skupinou skriptů naprogramovaných v Perlu. Když proběhlo sjednocení PHP s Form Interpreter, vzniklo spojení PHP/FI, v kterém je implementace pro programovací jazyky C. Navíc obsahuje komunikaci s databází, což způsobilo vývoj prvních dynamických aplikací na internetu. Veřejně byl PHP uvolněn na začátku července 1995 z důvodů zdokonalení překladače kódu php hledáním nejrozličnějších chyb. Tato verze byla nazvána PHP 2. Jazyk se těšil velkému úspěchu mezi veřejností a tím i dalšímu vývoji. V roce 1997 proběhlo k přejmenování PHP zkratky na Hypertext Preprocesor. V dnešní době se používá verze PHP 5.2.9 a verze 5.3.0 je v RC verzi [10].

Výhody [10]

- Vyvinuto speciálně pro internetové aplikace,
- spousta funkcí již v základní knihovně PHP,
- výborná podpora většiny databázových systémů,
- multiplatformost (Linux, Mac-OS, Windows,...),
- standardní podpora ze strany poskytovatelů hostingových služeb,

- spousty projektů nebo kódů lze bezplatně použít (WordPress, phpBB a další),
- dobrá internetová dokumentace v mnoha jazykových mutacích,
- svobodná licence.

Nevýhody [10]

- PHP kód nelze přeložit do strojového kódu. Skript se musí při každém spuštění znovu přeložit,
- při zpracování požadavku neudrží kontext aplikace a kompiluje ji vždy znovu, což snižuje výkon.

3.1.4 AJAX

Ajax (Asynchronous JavaScript and XML) je technologický způsob vývoje dynamických aplikací, které mění svůj obsah bez nutnosti opětovného načítání [13]. Umožňují pohodlnější prostředí oproti klasickým webovým aplikacím, ale je potřeba používat nejmodernější webové prohlížeče pro správnou funkčnost aplikací. Ajaxové aplikace využívají především tyto technologie: [4]

- HTML, XHTML a CSS k zobrazení informací
- DOM a JavaScript pro dynamické změny zobrazovaných informací
- XMLHttpRequest pro asynchronní přenos dat webového serveru (většinou se využívá formát XML, ale je možno využít libovolný formát včetně HTML a prostého textu) [4].

Informace o technologii ajax poprvé zazněly v dubnu 2005 v článku „Jesse James Garrett“ s názvem „Ajax: A New Approach to Web Applications“ (*Ajax: Nový přístup k webovým aplikacím*). Myšlenky, na kterých je založen jsou, mnohem starší. Mezi počátky lze považovat IFRAME v Microsoft Internet Explorer 3.0 nebo později LAYER od Netscape Navigator 4.0, od kterého se upustilo v roce 1997, kdy se začala vyvíjet Mozilla Macromedia Flash, která také zvládá asynchronně předávat data serveru. V roce 1998 přišel Microsoft s novou službou nazvanou Remote Scripting, u které v prohlížeči běžela Java applet, který komunikuje se serverem a nabízí služby JavaScriptovým funkcím. To bylo implementováno od verze Microsoft Internet Explorer 4.0 a od Netscape Navigator 4.0. Dále od verze Microsoft Internet Explorer 5.0 byl zaveden objekt XMLHttpRequest. O rozšíření ajaxových služeb a získání velké popularity se postaraly především nové aplikace od společnosti Google (Gmail, Google Maps, atd.) [10].

XMLHttpRequest nabízí rozhraní, pomocí kterého webové aplikace komunikují mezi serverem a prohlížečem přes protokol HTTP. V současnosti se W3C zabývá přípravou specifikací XMLHttpRequest, která by měla sjednotit chování všech prohlížečů.

3.1.4.1 JavaScript

JavaScript je multiplatformní objektově orientovaný skriptovací jazyk [17]. O jeho vývoj se zasloužil především Brendan Eich ze společnosti Netscape. Nyní se využívá jako programovací jazyk pro www stránky, který se vkládá přímo do kódu HTML. JavaScript běží na straně klienta, tj. přímo ve vašem počítači na rozdíl od jiných programovacích jazyků, jako např. PHP, ASP NET, které, jak jsme si již výše řekli, pracují na straně serveru, což však způsobuje bezpečnostní omezení. JavaScript nemůže pracovat se všemi komponentami jako jiné programovací jazyky. Jde především o souborové systémy, aby se zajistilo soukromí uživatelů. Javascript ve většině případů ovládá prvky GUI (tlačítka, textová políčka, různá upozornění pomocí vyskakovacích zpráv, atd.) [10].

Javascript patří do rodiny C/C++/Java. S programovacím jazykem Java však nemá

téměř nic společného, tedy kromě názvu a podobné syntaxe. Byl v červenci 1997 standardizován organizací ECMA (European Computer Manufacturers Association) a v srpnu 1998 organizací ISO (International Organization for Standardization). JavaScript byl obchodním označením implementace firmy Netscape, který byl vyvíjen pod názvem Mocha, později LiveScript. Společně s firmou Sun Microsystems byl v roce 1995 použit jako doplněk k jazykům HTML a Java [5].

3.1.4.2 Document Object Model

Document Object Model (DOM – objektový model dokumentu) je objektově orientované zobrazení XML nebo HTML dokumentu. DOM je API nabízející cestu ke změně obsahu, struktury nebo části. Dříve měl každý prohlížeč své jedinečné svérázné rozhraní k zacházení s HTML elementy pomocí JavaScriptu. Nekompatibilita www stránek mezi různými prohlížeči však způsobila zavedení W3C Document Object Model (W3C DOM). Předchozí specifické prostředí bylo nazváno Intermediate DOM (*DOM*) [10].

3.1.5 Databázové systémy

Nejčastější způsob využití počítačů je v oblasti zpracování a uchovávání dat dříve nazývané jako hromadné zpracování dat. Tyto systémy se nazývají informační systémy, které slouží k zpracování, uchovávání a vyhledávání uložených dat [6].

Problémy hromadného zpracování dat vedly k vývoji databázových systémů a později k relačním databázím, které mají následující vlastnosti:

- Datové struktury souborů jsou odděleny od uživatelských programů,
- data jsou přístupná jen prostřednictvím programu databázového systému,
- způsob vyhodnocení dat není omezen,
- je zajištěn paralelním přístupem více uživatelů současně a jejich bezpečnost [6].

Data v relačních databázích již nejsou v izolovaných souborech, ale jsou uloženy v centrálně zpracovávané struktuře, nazvané databáze nebo také báze dat, která umí s daty i pracovat, tj. systém řízení báze dat. Databáze je schopná pracovat paralelně a rychlým způsobem odpovídat na různé požadavky. Ulehčují zpracování různých dat od soukromých údajů v nemocnicích přes databáze obchodů nebo firem až k informačním systémům různých organizací, např. školní informační systém apod. [6].

3.1.6 Dotazovací jazyk SQL

Dotazovacím jazykem SQL jsme se již zabývali v kapitole 2.3.2.1, zde jen uvedeme doplňující údaje o tomto jazyku.

Během 70. let 20. století se firma IBM zabývala výzkumem relačních databází. Bylo potřeba vyvinout jednoduchý jazyk, který by se podobal přirozenému jazyku (angličtině) pro ovládání těchto databází. Vznikl jazyk SEQUEL (Structured English Query Language) [10].

K vývoji jazyka se připojily další firmy, v roce 1979 uvedla na trh firma Relation Software Inc. (dnešní Oracle Corporation) svojí platformu Oracle Database. IBM představila v roce 1981 systém SQL/DS a v roce 1983 systém DB2. Ve všech systémech se využívalo dotazovacího jazyka SEQUEL, který byl později přejmenován na SQL [10].

S rostoucím využíváním databází bylo potřeba standardizovat dotazovací jazyk. Společnost ANSI měla v úmyslu vydat jako standard zcela nový jazyk RDL, ale SQL se natolik prosadil, že byl přijat jako standard SQL-86 podle roku, kdy byl přijat. Následně

se přišlo na nedostatky a došlo k přijetí standardu SQL-92 a později SQL-99. Nejnovějším standardem, který reaguje na nejmodernější databáze je z roku 2008. V RDBMS se většinou používá SQL-99 nebo 2003 [10].

Standards podporují všechny relační databáze, které jsou dnes na trhu, avšak databáze některých firem obsahují rozšíření, která nejsou obsažena ve standardu, a proto přenositelnost SQL dotazů mezi jednotlivými databázemi je omezená [10].

3.1.6.1 MySQL

MySQL je databázový systém vytvořený švédskou firmou MySQL AB. Mezi hlavní autory patří Michael „Monty“ Widenius a David Axmark. MySQL je multiplatformní relační databázi, která komunikuje s uživatelem pomocí jazyka SQL [12]. Stejně jako u jiných databází se jedná o implementaci standardu nejnovějšího SQL s dalšími rozšiřujícími prvky společnosti [10].

MySQL lze jednoduše implementovat a instalovat na různé operační systémy, jako např. Linux, Microsoft Windows, Mac-OS, apod. Díky svobodnému šíření software, svému výkonu a snadné implementaci patří v současné době k nejpoužívanějším relačním databázím na internetu. Nejoblíbenější kombinace software jako základní výbava pro webový server je: PHP, MySQL a Apache. Je nabízena ve dvou licencích [10]:

- bezplatná licence GPL,
- komerční placená licence.

3.2 Přístupnost webových stránek

Při vývoji internetových aplikací bychom měli myslet na to, že internet nemusí prohlížet jen zdraví lidé nebo lidé se stejným technickým vybavením. Ředitel konsorcia W3C a zakladatel služby www, Tim Berners-Lee, vyslovil větu, která nejlépe vystihuje přístupnost webových stránek: „Síla webu je v jeho univerzalitě. Přístup pro každého nezávisle na schopnostech je jeho základní prvek“ [5]. Jednou větou lze přístupnost webových stránek charakterizovat: „Přístupná webová stránka je použitelná pro každého uživatele Internetu a to nezávisle na jeho postižení, schopnostech, znalostech, zkušenostech či zobrazovacích možnostech“ [5].

Z výše uvedených informací lze usoudit, že při vývoji internetových aplikací musíme dodržovat určité zásady přístupnosti webových stránek a to zejména, aby aplikace byly přizpůsobeny dobré čitelnosti pro slabozraké nebo uživatele s černobílým monitorem a u obrázků psát popisky nebo zpřístupnit informace z grafů v prostém textu pro uživatele, kteří používají hlasové odečítače obrazovky. V neposlední řadě, v případě použití rozbalovacího menu pomocí JavaScriptu, navrhnout toto menu tak, aby bylo dobře ovladatelné i bez JavaScriptu a využívat Flash na stránkách jen jako doplňující grafiku [9].

Možná se zdá, že je to zbytečná práce navíc. Není tomu tak, protože nejvíce hendikepovanými uživateli jsou zejména vyhledávací roboty, které prochází internet a indexují obsah stránek, jsou schopni si ale poradit jen s prostým textem. Jinými slovy řečeno, pokud máme stránky např. ve flashi, tak robot takovou stránku přeskočí, neboť si s takovým obsahem neporadí a stránky si nezaindexuje do své databáze, protože vyhodnotí stránku jako nezajímavou, což způsobí horší vyhledávání těchto stránek na internetu [9].

4 EXPERTNÍ SYSTÉM FUZZY NOTEBOOKŮ

V následující kapitole se dozvíme, jakým způsobem byl v naší aplikaci navržen rozhodovací *expertní systém pro e-shop s notebooky*. Tento systém nám zajistí popis technických informací pomocí přirozeného jazyka tak, aby si každý člověk mohl určit jaké požadavky má na svůj nový notebook. Protože notebooky se neustále vyvíjí a to co dnes je „rychlý notebook“, za rok už bude „pomalý notebook“, tak nelze tento rozhodovací systém naprogramovat napevno, ale musíme nějakému expertovi, který se bude starat o expertní systém, umožnit namodelovat a jednoduše editovat všechny parametry, které byly již dříve určeny. Při návrhu modelovacího prostředí byl kladen požadavek, aby tento systém byl jednoduše ovladatelný. Kromě naprogramovaného modelovacího systému jsou součástí této aplikace již namodelované znalostní informace, které byly navrženy v modelovacím prostředí aplikace. Některé části tohoto modelu by mohly být navrženy znovu a lépe, ale to už je na uvážení experta, který je pověřený k udržování aktuálního znalostního modelu s dnešními trendy a novinkami notebooků.

Nyní si definujme, co je to expertní systém. Jak si lze intuitivně odvodit, jde o počítačový program, v našem případě *www* aplikaci, která díky znalostem získaných od experta, je schopná pomocí zadaných neurčitých údajů rozhodnout, který notebook by byl pro zákazníka nejvhodnější. V podstatě tento systém umožňuje i na internetu nabídnout zákazníkům technickou podporu, jakou má zákazník při nákupu v klasickém obchodě [8].

Pro zpracování neurčitých údajů jsme použili fuzzy logiku, která nám nabízí výborné rozhodování a na základě evaluačních výrazů přirozeného jazyka je schopná si vypočítat míru ohodnocení ke konkrétním parametrům notebooků, a následně dokáže na základě tohoto ohodnocení rozhodnout, jestli daný notebook ještě vyhovuje požadavkům zákazníka nebo již ne.

4.1 Návrh tabulek

Informace, které získáme od experta, si musíme nejprve někam uložit, aby byl systém schopný tyto informace na základě požadavků uživatelů vyhledat. Především potřebujeme tabulku, ve které budeme mít uloženy konkrétní údaje, které podrobně popíší všechny komponenty notebooku. Tato tabulka je základní tabulkou každého internetového obchodu. Kromě této tabulky si musíme navrhnout tabulku, kam si budeme ukládat lingvistické proměnné, které nám poslouží k popisu notebooků v přirozeném jazyce. Poslední tabulka nám poslouží k uložení ohodnocení notebooků na základě předchozích dvou tabulek.

4.1.1 Tabulka notebooků

Následující tabulka nám slouží pro určení, zda je notebook rychlý, pomalý, apod. V kapitole 5 bude reprezentováno využití dat z tabulky pro katalog notebooků.

Tab. 2 notebook.

Název sloupce	Typ sloupce	Popis
ID	int(10)	Slouží k identifikaci daných notebooku
ZNACKA	varchar(32)	Jméno značky notebooku
OZNACENI	varchar(32)	Označení typu notebooku
FOTO	mediumblob	Fotka notebooku pro snadné zobrazení
TYP	varchar(18)	Typ daného obrázku (jpeg, png, atd.)
OPERACNI_SYSTEM	varchar(40)	Název operačního systému v notebooku
PROCESOR	varchar(35)	Název procesoru v notebooku
FREKVENCE_PROCESORU	double	Číselná hodnota frekvence v procesoru
CHIPSET	varchar(24)	Název chipsetu v notebooku
PAMET	varchar(5)	Číselná hodnota velikosti paměti
KAPACITA_DISKU	int(11)	Velikost disku v notebooku
OTACKY_DISKU	int(11)	Otáčky disku v notebooku
MECHANIKA	varchar(25)	Název optické mechaniky v notebooku
VELIKOST_DISPLEJE	double	Velikost displeje notebooku v palcích
ROZLISENI_DISPLEJE	varchar(15)	Rozlišení displeje v notebooku
GRAFICKA_KARTA	varchar(32)	Název grafické karty v notebooku
VSTUP_SLUCHATKA	enum('1','0')	Možnost připojení sluchátek
VSTUP_LINEIN	enum('1','0')	Možnost připojení Line-in
VSTUP_MIKROFON	enum('1','0')	Možnost připojení mikrofону
VYSTUP_ANALOG	enum('1','0')	Možnost analogového výstupu
VYSTUP_DIGITAL	enum('1','0')	Možnost digitálního výstupu
VYSTUP_TV	enum('1','0')	Možnost televizního výstupu
VYSTUP_HDMI	enum('1','0')	Možnost HDMI výstupu
REPRODUKTORY	enum('1','0')	Integrované reproduktory v notebooku
MIKROFON	enum('1','0')	Integrovaný mikrofón v notebooku
WEBKAMERA	varchar(4)	Integrovaná webkamera, udávající se MPx
FAX	varchar(20)	Integrovaný fax-modem
SIT	enum('1','0')	Ethernet
WIFI	enum('1','0')	Bezdrátové připojení k internetu
BLUETOOTH	enum('1','0')	Připojení zařízení pomocí bluetooth
INFRAPORT	enum('1','0')	Připojení zařízení pomocí infraportu
USB	int(11)	Počet USB v notebooku
FIREWARE	enum('1','0')	Připojení zařízení pomocí firmware
CARD_P	enum('1','0')	Možnost vložení PCMCIA karet
CARD_E	enum('1','0')	Možnost vložení ExpressCard
CTECKA_KARET	varchar(50)	Integrovaná čtečka karet v notebooku
TOUCHPAD	enum('1','0')	Integrované ovládání myši
BATERIE	varchar(12)	Počet článků v baterii
ROZMERY	varchar(20)	Rozměry notebooku
HMOTNOST	double	Hmotnost notebooku
CENA	int(11)	Cena notebooku v českých korunách

Z výše uvedené tabulky lze usoudit, že obsahuje všechny důležité technické informace, kterými můžeme popsat notebook. Všechny tyto informace jsou přístupné v přehledném katalogu notebooků, který si podrobněji popíšeme v následující kapitole 5. Všechny výše navržené řádky tabulky lze intuitivně pochopit pomocí jejich popisu, proto vysvětlíme podrobněji jen položky, jejichž jméno není dostačující k pochopení jejich

významu. Řádek FOTO uvádíme v tabulce z důvodu, že nejjednodušším způsobem, jak u daného typu notebooku ukládat i jeho obrázek, je uložit si ho přímo do databáze. Následující řádek obsahuje TYP fotografie, protože fotky jsou většinou různého formátu (jpg, bmp, png, apod.), proto si do databáze uložíme i jejich formát, na základě kterého jsme pak schopni obrázek načíst zpět z databáze. Nyní si popíšme datové typy, které máme u jednotlivých řádků:

- ✓ **Int(číslo)** – jedná se o typ integer s maximálním počtem znaků uvedeným v závorce,
- ✓ **varchar(číslo)** – string s maximálním počtem znaků uvedeným v závorce,
- ✓ **mediumblob** – formát pro ukládání obrázku do databáze,
- ✓ **double** – reálné číslo s dvojnásobnou přesností,
- ✓ **enum('1','0')** – jedná se o formát, kde 1 znamená ANO a 0 – NE, jiné hodnoty nejsou přípustné.

4.1.2 Tabulka lingvistických proměnných

Do následující tabulky budeme ukládat lingvistické proměnné k různým parametrům notebooků. Je zřejmé, že pro každý parametr bude několik lingvistických proměnných, např. pro velikost notebooků budou lingvistické proměnné nejmenší, menší, střední, větší a největší, pro rychlost notebooku to bude např. nejrychlejší, rychlý, střední, pomalý, apod. U každé lingvistické proměnné musí být definována funkce a interval, pro který definujeme lingvistickou proměnnou.

Tab. 3 *lingvistic*.

Název sloupce	Typ sloupce	Popis
ID_LIN	int(10)	Identifikace lingvistické proměnné
HARDWARE	varchar(25)	Identifikace typu hardware, pro který je definovaná lingvistická proměnná
NAZEV	varchar(30)	Název lingvistické proměnné
OD	varchar(30)	Začátek funkce lingvistické proměnné
STRED	varchar(30)	Střed funkce lingvistické proměnné
DO	varchar(30)	Konec funkce lingvistické proměnné
FUNKCE	enum('1','2','3')	Definice funkce lingvistické proměnné 1=minimum, 2=střed, 3=maximum

Výše uvedené řádky v tabulce jsou intuitivně pochopitelné z popisu v tabulce a datové typy používané v této tabulce jsme si již výše vysvětlili.

4.1.3 Tabulka fuzzy hodnot

Následující tabulka je logickým vyústěním našeho návrhu. Jak se dozvíme při popisu relací, tato tabulka slouží k propojení předchozích dvou tabulek a navíc se zde ukládají míry neurčitosti, pomocí kterých se expertní systém rozhoduje, jestli daný notebook ještě do dané množiny patří nebo nepatří. V této tabulce jsou uloženy všechny rozhodovací informace a tím pádem lze laicky obrazně říci, že jde o hlavní mozek našeho expertního systému.

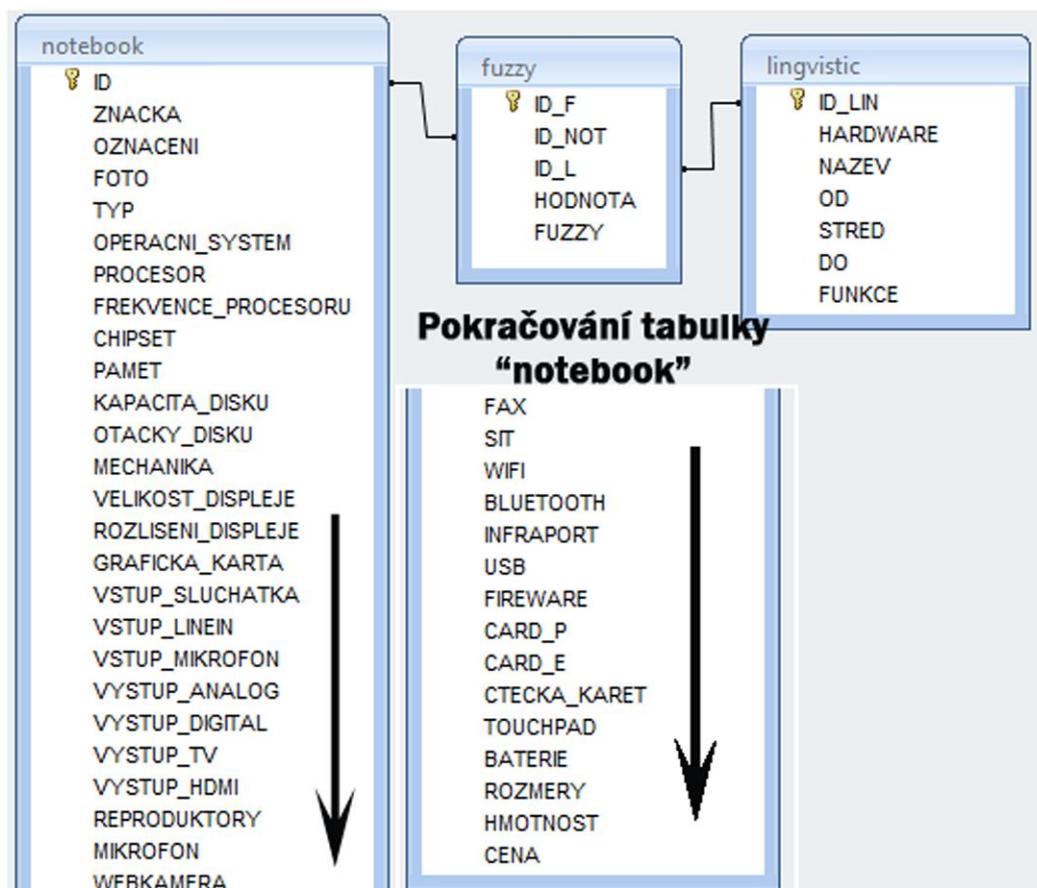
Tab. 4 fuzzy.

Název sloupce	Typ sloupce	Popis
ID_F	int(10)	Identifikace fuzzy hodnoty
ID_NOT	int(10)	Identifikace daného notebooku
ID_L	int(10)	Identifikace dané lingvistické proměnné
HODNOTA	varchar(35)	Proměnná notebooku, která se hodnotí
FUZZY	varchar(35)	Fuzzy ohodnocení na základě lingvistické proměnné

Řádek ID_NOT zde používáme k navázání s tabulkou notebook a jejím parametrem ID. Dále řádek ID_L používáme k propojení s tabulkou lingvistic a jejím parametrem ID_LIN.

4.1.4 Relace

V předchozí kapitole jsme narazili na problém propojení tabulek, do kterých si ukládáme data pro expertní systém. Z Obr. 12 i předchozího textu je zřejmé, že nejdůležitější tabulkou našeho expertního systému je tabulka s názvem fuzzy, ve které jsme si uložili na základě předchozích kroků ohodnocení daných parametrů a jejich lingvistických popisů.



Obr. 12 Relace mezi tabulkami.

4.2 Lingvistické proměnné

O lingvistických proměnných již toho bylo v tomto textu napsáno mnoho, nyní se však blížíme k samotnému návrhu a implementaci proměnných v našem expertním systému. Jak již bylo naznačeno v kapitole 2.2, lingvistické proměnné slouží k modelování různých problémů pomocí přirozeného jazyka, tj. zavádíme do konkrétních technických parametrů neurčitosti, které nabízí uživatelům snadnější definici svých požadavků. Nyní si projdeme, které parametry a vlastnosti notebooků jsme se rozhodli použít k modelování neurčitosti, jaké lingvistické proměnné je nejvhodnější použít a jakým způsobem lze v naší aplikaci modelovat lingvistické proměnné.

4.2.1 Výběr parametru notebooků

Nyní je potřeba navrhnout vlastnosti a komponenty notebooků, které budou zahrnuty do fuzzy modelu. Při nákupu notebooku se řídíme několika parametry, které jsou pro nás nejdůležitější. Samozřejmě pro každého člověka mohou být dané parametry jiné, proto je potřeba navrhnout takové prvky, které budou nejlépe vyhovovat většině uživatelů. Byly vybrány tyto parametry notebooků, na základě kterých se mohou uživatelé rozhodnout, které parametry jsou pro ně důležité:

- ✓ **Značka** – Během posledních let si uživatelé notebooky velmi oblíbili a z komerčních důvodů se některé firmy snaží nabídnout své notebooky za co nejnižší cenu. To se však odráží na kvalitě provedení. Některé firmy si na kvalitě svých výrobků zakládají. Z předchozího důvodu jsme vybrali značku pro definování, jestliže jde uživateli o kvalitu notebooku.
- ✓ **Velikost displeje** – Vývoj notebooku způsobil postupnou miniaturizaci komponent notebooku a tím i zmenšení rozměrů notebooků, což umožňuje lepší mobilitu, která je hlavní výhodou notebooků.
- ✓ **Procesor** – Je základním mozkiem notebooků. Čím kvalitnější a rychlejší procesor, tím bude notebook rychlejší.
- ✓ **Pevný disk** – Je základním úložištěm pro operační systém, data a programy uživatele v notebooku. V dnešní době je trend mít k dispozici co největší pevný disk.
- ✓ **Operační paměť** – S rychlostí úzce souvisí i operační paměť. Čím větší velikost paměti, tím máme více prostoru pro spouštění aplikací, s kterými pracujeme.
- ✓ **Grafická karta** – Úzce souvisí s aplikacemi, které jsou velmi náročné na grafiku, tj. grafické programy a počítačové hry.
- ✓ **Baterie** – Souvisí s mobilitou notebooků, protože nemáme vždy možnost připojit notebook do elektrické energie a proto požadujeme, aby notebook vydržel nejnutnější dobu pracovat na baterii.
- ✓ **Hmotnost** – Úzce souvisí s velikostí notebooků a jeho mobilitou. Čím je menší notebook, tím je dosaženo i menší hmotnosti.
- ✓ **Cena** – Úzce souvisí se všemi výše uvedenými parametry a především je závislá na kvalitě provedení notebooku a jeho velikosti. Většinou máme omezený počet financí, které máme k dispozici pro nákup notebooků.

Výše uvedené hardware a vlastnosti notebooku jsou popsány lingvistickými proměnnými, na základě kterých se vygenerují množiny fuzzy čísel, které ohodnocují, s jakou mírou daný prvek patří do dané lingvistické proměnné. Pro ilustraci je na Obr. 13

stránka aplikace s nastavením lingvistických proměnných a zcela shodná základní stránka pro generování fuzzy ohodnocení pro výše uvedené parametry notebooků.



Obr. 13 Obrazovka nastavení lingvistických proměnných pro dané parametry.

Na závěr je potřeba říci, že výše uvedené parametry, které lze popsat lingvistickými proměnnými, jsou pevně dané a není možné se do budoucna rozhodnout, že požadujeme modelovat ještě jiný parametr. V tomto případě bychom museli tento parametr naprogramovat. Věříme však, že výše uvedené parametry jsou jedny z nejdůležitějších k výběru nového notebooku pro všechny uživatele a navržený omezený počet parametrů bude pro naši aplikaci dostačující.

4.2.2 Modelování fuzzy lingvistických proměnných

Implementace *modelování fuzzy lingvistických proměnných* je provedena pomocí formuláře, který slouží k samotnému modelu a současně je tento formulář doplněn obrázkem, který vygeneruje PHP na základě zadaných informací přes formulář do databáze. Tyto výše definované vlastnosti jsou zobrazované v přehledném seznamu pro jednotlivé parametry nebo hardware notebooků. Podrobněji se implementací jednotlivých částí aplikace budeme věnovat v další části textu.

Již máme navrhnuť, který hardware a vlastnosti notebooků budeme popisovat lingvistickými proměnnými a můžeme se zabývat návrhem znalostního modelu, kterému je nutno věnovat velkou pozornost, protože jde o klíčovou vlastnost každého expertního systému. Pokud navrhne nevhodně znalostní model, expertní systém nebude schopen správně rozhodovat na základě požadavků uživatele. V případě dobrého návrhu modelu expertní systém začne tzv. „žít svým životem“ a my budeme jen minimálně zasahovat do jeho modelu. Před začátkem modelování by si měl expert promyslet, jaký je nejvhodnější popis daného parametru a který interval bude nejvhodnější pro danou lingvistickou proměnnou. Na základě tohoto modelu jsme schopni určit míru ohodnocení všech parametrů popsaných lingvistickými proměnnými, které jsme si definovali v předchozí kapitole a které zároveň slouží k snadnějšímu porozumění uživatelem daného parametru na základě využití intuitivního neurčitého popisu. U každé lingvistické proměnné lze definovat tři tvary fuzzy množin: minimální, střední a maximální funkce. Tyto funkce jsme již definovali v kapitole 2.2. Nyní si popíšeme, jaké lingvistické proměnné jsou navrženy v našem modelu:

- ✓ **Značka** – nejkvalitnější, kvalitní, střední, neutrální, levnější,
- ✓ **Velikost displeje** – největší, větší, normální, menší, nejmenší,
- ✓ **Procesor** – nejrychlejší, rychlý, průměr, pomalý, nejpomalejší,
- ✓ **Pevný disk** - nejvíce místa, hodně místa, průměrně místa, málo místa, nejméně místa,
- ✓ **Operační paměť** – nejvíce, střední, nejméně,
- ✓ **Grafická karta** – pro práci, hry,
- ✓ **Baterie** – velká, průměrná, malá,
- ✓ **Hmotnost** – nejlehčí, lehčí, průměr, těžší, nejtěžší,
- ✓ **Cena** – nejdražší, dražší, střední, levnější, nejlevnější.

Protože modelování fuzzy lingvistických proměnných je u všech parametrů analogické, budeme se v dalším výkladu zabývat jen jedním parametrem. Jako výchozí parametr, který podrobněji v následujícím textu popíšeme, jsme vybrali „Velikost displeje“.

4.2.2.1 Návrh lingvistických proměnných

V případě, že již máme promyšlené všechny výše uvedené lingvistické proměnné, můžeme přistoupit k zavedení těchto hodnot do našeho expertního systému. Pro snadnější modelování se na každé stránce zobrazuje interval daného parametru, takže tyto hodnoty máme stále před sebou a tak můžeme snadno vyplnit údaje, které po nás formulář požaduje.

V první fázi si musíme uvědomit, kterou funkci vyžadujeme modelovat. V případě, že požadujeme modelovat např. nejmenší velikost notebooku, zvolíme minimální funkci, analogicky pro největší velikost notebooku zvolíme maximální funkci,

v ostatních případech zvolíme střední funkci a vybereme tlačítko vybrat, viz Obr. 14. Po zvolení tlačítka se nám zobrazí formulář, viz Obr. 15, kam zadáme námi zvolené údaje. Některé informace v tomto formuláři jsou předvyplněné z důvodu snadnější práce s formulářem a následným zpracováním zadaných informací do výsledného výpisu.

Obr. 14 Výběr funkce fuzzy množin.

Nyní si popíšeme podrobněji políčka našeho formuláře pro vložení nové lingvistické proměnné. Následující výklad je shodný i s formulářem pro editaci:

- ✓ **Hardware** – toto políčko je předvyplněné a nelze je měnit. Slouží k identifikaci, který parametr notebooku modelujeme,
- ✓ **Název lingvistické proměnné** – zde napíšeme, jak se naše lingvistická proměnná bude jmenovat, např. „nejmenší“,
- ✓ **Interval od** – zde napíšeme hodnotu, kde tato hodnota a hodnoty, které jsou větší nebo menší od této hodnoty, budou vždy ohodnoceny „1“. U minimální a maximální funkce bude toto políčko vždy předvyplněné hodnotou „n“,
- ✓ **Interval střed** – zde napíšeme hodnotu, od které se začne ohodnocení na základě zvolené funkce snižovat nebo zvyšovat,
- ✓ **Interval do** – zde napíšeme hodnotu, kde tato hodnota a hodnoty, které jsou větší nebo menší od této hodnoty, budou vždy ohodnoceny „0“,

Aby byly dobře vypočteny míry neurčitosti, je nutno do políček intervalu od, střed a do vkládat pouze číselné hodnoty.

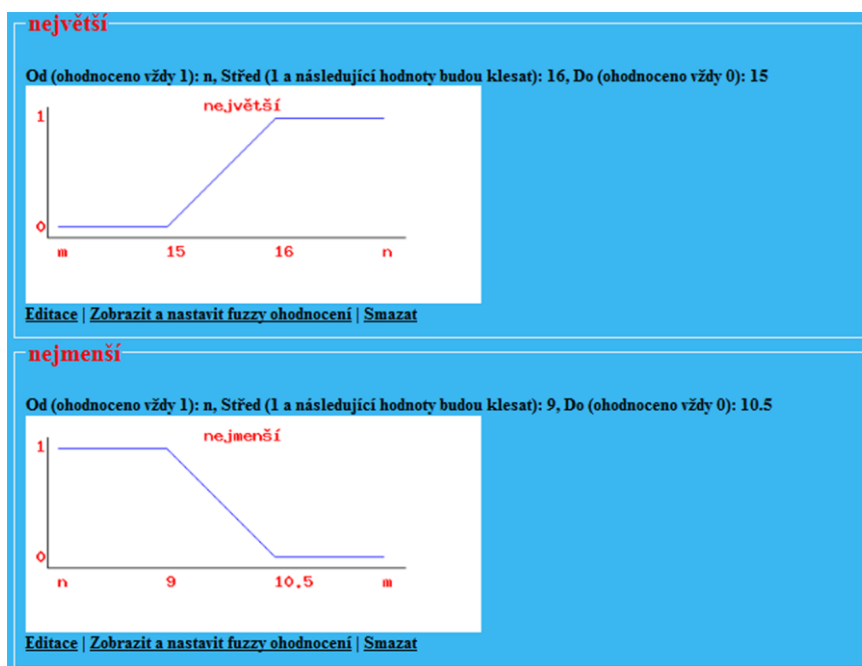
Obr. 15 Vložení nové lingvistické proměnné.

Jakmile odešleme formulář zvolením tlačítka Uložit, tak se nám vytvoří nová lingvistická proměna k danému parametru, v našem případě k velikosti notebooků, která se

zařadí do množiny ostatních proměnných tohoto parametru. Následujícím způsobem můžeme vytvořit tolik lingvistických proměnných, kolik uznáme za vhodné.

4.2.2.2 Model lingvistických proměnných

Následující Obr. 16 znázorňuje jen část modelu lingvistických proměnných. Kompletní model velikosti notebooků ve větším rozlišení nalezneme v příloze na Obr. 44.



Obr. 16 Model lingvistických proměnných pro velikosti notebooků.

Implementace výše zmíněných částí aplikace je v podstatě jen práce s databází, tj. vkládání dat do databáze a následné načtení dat z databáze pomocí SQL jazyka, kde v první fázi si PHP vygeneruje obrázek zvolené funkce a tento obrázek popíše daty z databáze. Následně vygeneruje popisky a názvy lingvistických proměnných a tento proces provede pomocí cyklu tolikrát, kolik je v databázi uloženo lingvistických proměnných s výchozím parametrem „VELIKOST_DISPLETE“, který je předvyplněný v kolonce HARDWARE při vytváření nové lingvistické proměnné. Na závěr každé proměnné doplní odkazy pro editaci, smazání a přejítí ke generování fuzzy ohodnocení.

4.3 Generování fuzzy ohodnocení na základě lingvistických proměnných

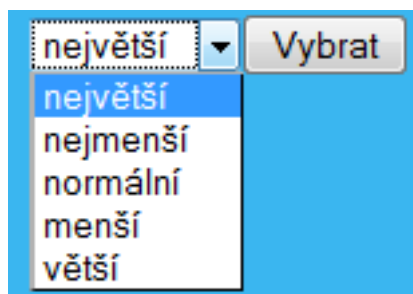
Generování fuzzy ohodnocení na základě lingvistických proměnných je implementováno tak, že v úvodu se zobrazí název lingvistické proměnné a vygenerovaný obrázek pomocí PHP s jeho nejdůležitějším popisem, aby uživatel měl přehled, pro jakou lingvistickou proměnnou generuje fuzzy ohodnocení. Samotné generování fuzzy ohodnocení je řešeno pomocí implementace funkcí pro výpočet ohodnocení, ke kterým se pro jejich důležitost podrobněji budeme zabývat v samostatné kapitole v dalším výkladu. Generování stránky je řešeno nejprve použitím výpočetních funkcí a zobrazení tabulky s vypočtenými hodnotami. Následuje tlačítko, které uloží ohodnocení do databáze. V poslední části

je tabulka, která zobrazuje notebooky a především jejich parametry a vygenerované ohodnocení, které se načítá z databáze. Součástí této stránky je také automatická kontrola ohodnocení všech notebooků u dané lingvistické proměnné. Pokud by náhodou nějaký notebook a jeho parametr u dané lingvistické proměnné neměl své ohodnocení, pak by nás na to naše aplikace upozornila.

Nyní, když již máme zpracována konkrétní data notebooků, která máme popsána pomocí lingvistických proměnných, můžeme přistoupit k samotnému ohodnocení hardware a vlastností notebooků. V našem expertním systému je ohodnocení vypočítáváno automaticky podle intervalu, který jsme si definovali u lingvistických proměnných. Naším úkolem je v této části si ověřit, jestli námi navržený interval v lingvistických proměnných odpovídá skutečnosti, tj. jestli hardware a vlastnosti notebooků jsou správně ohodnoceny. Proto také u každé lingvistické proměnné je odkaz s názvem „Zobrazit a nastavit fuzzy ohodnocení“. Tento odkaz nám poslouží k rychlé kontrole správného nastavení intervalu.

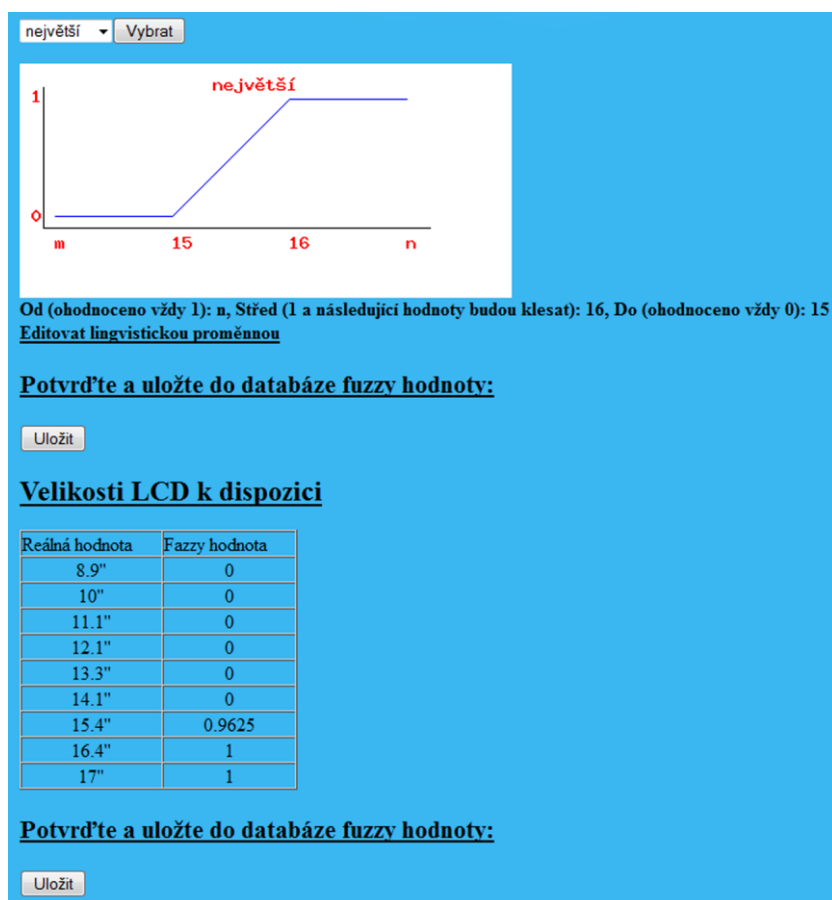
U lingvistické proměnné jsme si definovali interval, na kterém daná proměnná je platná, tj. definovali jsme si, které hodnoty patří do dané proměnné na 100% a také, které nepatří na 100%. Kromě těchto absolutních hodnot máme však i další části intervalu, u kterého nemůžeme rozhodnout, jestli do daného intervalu patří, přesněji řečeno jde o část intervalu od střední hodnoty až po konec intervalu. Následující hodnoty se musí dopočítat pomocí výpočetních funkcí v našem algoritmu expertního systému. Naším úkolem je především kontrola, jestli vypočtené hodnoty odpovídají realitě, kterou modelujeme, tj. aktuálním trendům výpočetní techniky. Naše aplikace je schopná ke každé lingvistické proměnné vygenerovat fuzzy ohodnocení a následně vyhledat všechny notebooky s určitými parametry, ke kterým se uloží do databáze jejich ohodnocení. Pomocí tohoto ohodnocení je aplikace schopná určit s jakou mírou daný notebook do určité lingvistické proměnné patří nebo nepatří.

Jakmile vybereme položku Generování fuzzy ohodnocení, zobrazí se nám stejná stránka, kterou jsme si popsali v kapitole 4.2.1. Zmiňovanou stránku si můžete prohlédnout na Obr. 13. Na této stránce si vybereme parametr, u kterého požadujeme zobrazit vygenerované fuzzy ohodnocení. Po výběru parametru následuje nenápadná stránka, na které je na výběr pouze možnost lingvistické proměnné (viz. Obr. 17), které jsme si namodelovali v předchozí kapitole. Jakmile si některou proměnnou zvolíme, zobrazí se nám komplexní rozhraní pro generování fuzzy ohodnocení.



Obr. 17 Výběr lingvistické proměnné u generování fuzzy ohodnocení.

Nyní si popíšeme, co se nám zobrazí po výběru dané lingvistické proměnné. Nejprve zde najdeme pro ilustraci obrázek dané funkce i s popisem aktuálních hodnot tak, jak jsme si je definovali v lingvistických proměnných. Následuje tabulka, kde jsou zobrazena vypočtená ohodnocení pro danou lingvistickou proměnnou (viz. Obr. 18).



Obr. 18 Výpočet fuzzy hodnot.

Algoritmus si v našem případě vyhledá všechny velikosti notebooků, co máme uloženy v tabulce „notebook“, a pomocí základních matematických operací implementovaných do výpočetních funkcí spočteme k těmto hodnotám fuzzy ohodnocení, které se zobrazí v již zmiňované tabulce. Je vhodné zde zmínit, že vypočtené hodnoty se automaticky neukládají do databáze. V případě, že jsme spokojeni s vypočtenými ohodnoceními, zvolme tlačítko uložit, které způsobí, že se k daným notebookům, které obsahují dané velikosti displeje, přiřadí vypočtené ohodnocení. Výše popsání části stránky si můžete prohlédnout na Obr. 18. V poslední části se nachází seznam notebooků a jejich ohodnocení na základě vypočtených hodnot. Tato tabulka je načtená z databáze a z důvodu úspory místa jsou zobrazovány jen notebooky s větším ohodnocením jak nula, protože notebooky s ohodnocením nula jsou pro nás nezajímavé a v podstatě u dané lingvistické proměnné nejsou potřebné. Následující tabulku si můžeme prohlédnout na Obr. 19. V tabulce ohodnocených notebooků jsou obrázky a jejich názvy pro rychlejší orientaci, o který notebook se jedná. Dále je tam sloupec hardware, který nás informuje, jaká velikost displeje je u daného notebooku a poslední informací je jejich fuzzy ohodnocení.

V aplikaci je implementována automatická kontrola ohodnocení všech parametrů notebooků. V případě, že některý notebook nemá své ohodnocení, vypíše se tato informace hned pod tabulku, ve které jsou vypočteny hodnoty ohodnocení. Tato funkce byla do aplikace přidána pro případy, kdyby zkolabovalo automatizované získávání fuzzy ohodnocení, které bude popsáno v následujícím výkladu.

	DELL Studio 1535	15.4	0.9625
	Sony VAIO NR31Z	15.4	0.9625
	Sony VAIO NS11S/S	15.4	0.9625
	Umax VisionBook M765T	15.4	0.9625
	DELL Latitude E5500	15.4	0.9625
	Sony VAIO FW21Z	16.4	1
	Acer Aspire 6935G	16.4	1
	HP EliteBook 8730w	17	1
	Lenovo THINKPAD W700ds	17	1
	Acer Aspire 7730G	17	1

Obr. 19 Tabulka notebooků a jejich fuzzy ohodnocení.

4.3.1 Výpočet fuzzy ohodnocení

Nyní si podrobněji popíšeme, jakým způsobem fungují naše výpočetní funkce ohodnocení v expertním systému. Matematický popis funkcí byl uveden v kapitole 2.1.2. Vypočtené hodnoty musí být co nejpřesnější, protože budou výchozí informací pro budoucí vyhledávač neurčitých informací. Pro tyto hodnoty byla zvolena čtyři desetinná místa, i když by stačilo pro naše účely i méně desetinných míst.

V následující části textu uvádíme ukázkou implementace v PHP kódu výpočetních funkcí ohodnocení v našem expertním systému. Dochází pouze k převedení matematických výrazů do programovacího jazyka, protože jde o základní znalosti programování, nebudeme je zde podrobněji popisovat.

- **Minimální funkce**

```
if ($funkce == "1" && $lcd <= $stred) {$fuzzy = "1";}
elseif ($funkce == "1" && $lcd >= $stred)
{$f = (($do-$lcd)/($do-$stred));
$fuzzy = round($f,5); }
if ($funkce == "1" && $lcd >= $do) {$fuzzy = "0";}
```

- **Střední funkce**

```

if ($funkce == "2" && $lcd <= $od) {$fuzzy = "0";}
elseif ($funkce == "2" && $lcd <= $stred)
{
    $f = (($lcd-$od)/($stred-$od));
    $fuzzy = round($f,5);}
elseif ($funkce == "2" && $lcd >= $stred)
{
    $f = (($do-$lcd)/($do-$stred));
    $fuzzy = round($f,5); }
if ($funkce == "2" && $lcd >= $do) {$fuzzy = "0";}

```
- **Maximální funkce**

```

if ($funkce == "3" && $lcd <= $do) {$fuzzy = "0";}
elseif ($funkce == "3" && $lcd <= $stred)
{
    $f = (($lcd-$od)/($stred-$od));
    $fuzzy = round($f,5); }
if ($funkce == "3" && $lcd >= $stred) {$fuzzy = "1";}
$pole[$lcd] = $fuzzy // fuzzy ohodnocení si uložíme do pole a
vyexportujeme do databáze

```

4.3.1.1 Automatizovaný výpočet fuzzy ohodnocení

Protože přiřazovat fuzzy ohodnocení každému novému notebooku by bylo velmi pracné, bylo potřeba implementovat automatizovaný výpočet fuzzy ohodnocení. Stejně tak po smazání notebooku je nutné smazat i jeho ohodnocení v ostatních tabulkách. Automatizované ohodnocení se provede v následujících místech:

- ✓ Při vložení nového notebooku proběhne automaticky ohodnocení jeho hardware a vlastností a následně se tyto informace uloží do tabulky „fuzzy“,
- ✓ při editaci notebooku se editují i ohodnocení notebooku.

Automatizovaný fuzzy výpočet byl navržen z důvodu snadnější práce s expertním systémem a také proto, jak se dozvíme v kapitole 5.3, vkládat nové notebooky nebo je upravovat může např. obsluha, prodavač a my musíme zajistit, aby se po vložení nového notebooku vypočetlo i ohodnocení. Na závěr zdůrazníme, že změnění nebo vytvoření nové lingvistické proměnné, jak již bylo v textu definováno, se míra ohodnocení automaticky v tabulce „fuzzy“ nezmění, protože se předpokládá, že o jádro expertního systému se bude starat expert, který si při návrhu lingvistických proměnných bude kontrolovat fuzzy ohodnocení, jestliže odpovídá realitě a v tomto případě zvolí tlačítko Uložit, čímž se změní data v databázi. Tento přístup se zdá výhodnější z důvodů nezávislosti znalostního systému na modelování experta, tj. databáze se aktualizuje, až je expert se svojí prací hotov.

4.3.1.2 Kontrola stejných řádků

Základním předpokladem pro správnou funkci expertního systému je, aby každý řádek v tabulce „fuzzy“ byl jedinečný. Asi by se nám nelíbilo, kdyby se stejný notebook vypisoval víckrát po sobě. Teoreticky by se nemělo stát, aby se v databázi objevily dva zcela identické řádky, ale i tak pro jistotu při modelování byla tato kontrola do aplikace přidána. Po zobrazení se nám vypíší řádky, které jsou identické s nějakým jiným řádkem.

V praxi se implementace provádí tak, že se kontroluje, jestli náhodou neexistují v databázi identické řádky s různým identifikačním číslem řádku, které musí být jedinečné. V případě, že takové řádky v databázi existují, tak se vypíší oba pro kontrolu, že jsou opravdu tyto řádky identické. My pak máme možnost jeden řádek smazat.

4.4 Vyhledávání notebooků založené na fuzzy expertním systému

Znalostní expertní systém v e-shopu nám dává velké možnosti využití. Kromě samotného katalogu popsaného přirozeným jazykem pro uživatele nabízí vyhledávání pomocí výrazů přirozeného jazyka pro ty uživatele, kteří nedisponují dostatečnými technickými znalostmi o parametrech notebooků. Všechny výše definované parametry notebooků jsou zařazeny v přehledném vyhledávači (viz. Obr. 20), v kterém máme na výběr všechny lingvistické proměnné, které jsme si popsali v předchozí části textu. Pomocí nich můžeme specifikovat požadavky v přirozeném jazyce, co by měl notebook splňovat.

Nejprve je potřeba si říci, že z neurčitých příkladů nemůžeme získat přesné informace. Tato myšlenka je základem k pochopení funkce tohoto vyhledávače. Na začátku této práce jsme si řekli, že absolutní přesnost je prakticky nedosažitelná a v našem případě to platí dvojnásob, protože to, co je např. střední cena pro nás, nemusí být střední cenou pro někoho jiného, ale o tom jsme se již zmínili při modelování lingvistických proměnných s tím, že je potřeba model navrhnout tak, aby co nejlépe vyhovoval požadavkům většiny uživatelů a ti, co si myslí, že notebookům rozumí dostatečně, si mohou zvolit katalog pro experty. V dnešních internetových obchodech jsme zvyklí, že zadáme přesné požadavky, např. notebooky do 20 000,- Kč. Pro programátora takového vyhledávače je tento problém jednoduchou záležitostí. Zadá do svého kódu jednoduchý SQL příkaz a vyhledávání funguje zcela přesně. Však také není důvod, aby nefungovalo, vždyť vyhledávání přesných údajů z databáze je hlavní silou databázových systémů.

V našem případě je situace poněkud jiná. Vyhledávání neurčitých informací popsaných přirozeným jazykem, např. můžeme chtít „střední“ cenu a pak nejprve si musí skript zjistit, jaké notebooky patří do této lingvistické proměnné, a ty vypsat. Výše popsaný mechanismus by fungoval výborně pro jeden parametr vyhledávání. Protože vyhledávač má k dispozici devět možností, z kterých si může uživatel vybrat a specifikovat své požadavky a notebooky se ve většině případů neprodávají jako komponenty, ale jako sestavy, museli jsme vymyslet nějaký kompromis při vyhledávání. Pro tyto případy vyhledávání se využívá vícekritériálního rozhodování, které si podrobněji popíšeme v následujícím výkladu. Každý řádek vyhledávače si musí zjistit, co vlastně daná neurčitá informace znamená, a tyto notebooky si uložit do pomocné tabulky. Dále pokračuje následujícím řádkem a tyto notebooky se opět přidají do pomocné tabulky. Jakmile se projdou úplně všechny parametry, pomocí vícekritériálního rozhodování se z pomocné tabulky vyberou notebooky, které se následně vypíší uživateli. Také bylo potřeba vyřešit, co nastane, když uživatel nevybere některý z parametrů. Řešení se opírá o myšlenku, že když uživatel některý parametr nevybere, tak tento parametr pro něho není důležitý, tj. vyhledávací algoritmus vybere notebooky, které v tomto parametru mají stupeň příslušnosti větší než 0,5.

Snahou tohoto textu bylo vysvětlit, jakým způsobem vyhledávač funguje v případě, že zadáme např. požadavek na střední cenu notebooku a ostatní parametry nespecifikujeme. Pak to pro vyhledávač znamená, že je pro nás důležitá cena notebooku a dále notebooky se stupněm příslušnosti větším než 0,5. Z této charakteristiky vyplývá,

že ve výsledku klidně můžeme najít i notebook za 50 000,- Kč, i když nesplňuje střední cenu. Proč vyhledal algoritmus i notebooky, co nepatří do střední ceny? Protože logicky vzato algoritmus musel vybrat notebooky, které vyhovují lingvistickým proměnným „střední cena“, a u nedefinovaných parametrů omezit výběr na ty, které mají vyhovující stupeň příslušnosti, a to jak jsme již uvedli, volíme větší než 0,5. Na Obr. 20 je znázorněn formulář sloužící k vyhledávání notebooků pomocí expertního systému, z kterého můžeme na základě neurčitých informací vyhledat nejvhodnější notebooky.

Fuzzy vyhledávání notebooků

Kvalita značky notebooku:	vyberte ▼
Počítač na:	vyberte ▼
Velikost notebooku:	vyberte ▼
Rychlost notebooku:	vyberte ▼
Velikost disku:	vyberte ▼
Počet spuštěných programů:	vyberte ▼
Výdrž baterie notebooku:	vyberte ▼
Hmotnost notebooku:	vyberte ▼
Cena notebooku:	vyberte ▼

Obr. 20 Vyhledávací formulář pro expertní systém.

Obr. 21 znázorňuje vyhledávací dotaz a ukázkou informací, které vyhledávač pomocí neurčitých údajů našel. Pro rozsáhlejší výpis odkazují na přílohu Obr. 45.

Kvalita značky notebooku:	nejkvalitnější ▼
Počítač na:	práci ▼
Velikost notebooku:	menší ▼
Rychlost notebooku:	nejrychlejší ▼
Velikost disku:	hodně místa ▼
Počet spuštěných programů:	nejvíce ▼
Výdrž baterie notebooku:	velká ▼
Hmotnost notebooku:	lehký ▼
Cena notebooku:	střední ▼

Vašemu zadání vyhovují následující notebooky:




Asus F9E 1 2 3 >>

Skladem

Intel Core2 Duo T8100 2.1 GHz, paměť DDR2 2 GB, pevný disk 250 GB (5400 ot./min), velikost displeje 12.1", rozlišení displeje 1280x800, grafická karta integrovaná, síťová karta, wifi karta, bluetooth, SD, MMC, MS, MS-Pro, xD, mini SD, 2 kg

Cena: **19 590,- Kč**

Sony VAIO CS21S/R Skladem

Intel® Core™ 2 Duo T6400 2 GHz, paměť DDR2 4 GB, pevný disk 320 GB (5400 ot./min), velikost displeje 14.1", rozlišení displeje 1280 x 800, grafická karta NVIDIA® GeForce® 9300M GS, síťová karta, wifi karta, bluetooth, SD, MMC, MS, MS-Pro, xD, mini SD, 2.6 kg

Cena: **26 880,- Kč**

Obr. 21 Vyhledávací dotaz a výsledek jeho vyhledávání.

4.4.1 Pomocné tabulky k vyhledávání

Následující dvě tabulky slouží jako pomocné tabulky k vyhledávání zboží na základě lingvistických požadavků. Tyto tabulky úzce souvisí se zvoleným vícekritériálním rozhodováním. První tabulka, v programu nazvána *nejhorsí_vlastnosti*, slouží k nalezení nejhorších parametrů.

Tab. 5 *nejhorsí_vlastnosti*.

Název sloupce	Typ sloupce	Popis
ID_NEJHORSI	int(11)	Identifikace nejhorších vlastností
ID_NO	varchar(35)	Identifikace notebooku s nejhoršími vlastnostmi
MINFUZ	varchar(35)	Minimální fuzzy hodnoty

Druhá tabulka slouží k výběru všech notebooků s maximálním fuzzy ohodnocením nalezeným v tabulce *nejhorší_vlastnosti* a následný výběr notebooků, které se mají vypsat uživateli. Relace těchto tabulek nalezneme v následující kapitole 5.2.

Tab. 6 *pomoc*.

Název sloupce	Typ sloupce	Popis
ID_POMOC	int(11)	Identifikace řádku
LING	varchar(30)	Identifikace lingvistické proměnné
NOTEBOOK	varchar(30)	Identifikace notebooku
POCET	varchar(10)	Počet výskytů v pomocné tabulce

4.4.2 Vícekritériální rozhodování

V předchozí části textu jsme narazili na vícekritériální rozhodování [18]. Vyhledávač se skládá z devíti parametrů, podle kterých se musí rozhodnout, proto je potřebné využít některé z metod vícekritériálního rozhodování. Metody vícekritériální optimalizace lze rozdělit do dvou základních skupin. Podle způsobu zadání množiny přípustných variant, mluvíme o *vícekritériálním hodnocení variant*, druhým přístupem je tzv. *vícekritériální programování*. Podle zadaných informací o variantách a cílech uživatele můžeme agregovat větší počet kritérií pomocí vhodné skalarizující funkce. Pro upřesnění vyhledávání se mohou použít různé metody odhadu vah kritérií, které se stanovují z informací expertů, např. pomocí metody pořadí, anebo bodovací metody [18].

Vícekritériální hodnocení variant se dělí podle zadané úrovně kritérií, anebo významnosti kritérií. V našem případě jsme se rozhodli využít jednu z nejjednodušších metod, která se jmenuje vícekritériální programování. Při rozhodování můžeme dojít ke sporu, např. nelze výběrem získat notebook, který má špičkovou kvalitu a zároveň je nejlevnější. Tento rozpor požadavků zákazníka potřebujeme eliminovat a najít co nejlepší kompromis. V implementovaném expertním systému rozhodujeme na základě těchto předpokladů:

- Najdeme všechny notebooky, které mají v některém parametru nejnižší míru ohodnocení a ty si uložíme do tabulky *nejhorsí_vlastnosti*.
- Z této tabulky vybereme notebook, který má nejlepší hodnotu z hlediska agregace všech parametrů.
- V tabulce fuzzy pak vybereme notebooky, které mají hodnotu stejnou nebo vyšší než je hodnota určená v předchozím kroku a výsledné hodnoty vypíšeme.

- Jak již bylo uvedeno, pokud není zadána položka, vybereme všechny notebooky s ohodnocením této položky větším jak 0,5.

4.4.3 Implementace vyhledávání

Pro snadnější pochopení uvádím ukázkou vyhledávání v našem systému.

```
if ($slcd == 0)
{
$db->SQL_dotaz("SELECT DISTINCT ID_NOT FROM lingvistic,fuzzy where
ID_L=ID_LIN AND FUZZY>'0.5' AND HARDWARE='VELIKOST_DISPLEJE'");
while ($db->fetch_row())
{
$idnotebookux = $db->result("ID_NOT");
mysql_query("INSERT INTO pomoc (LING,NOTE)
VALUES('1000002','" . $idnotebookux . "')");
}}
```

- je vybrána položka, najdeme minimum ohodnocení a vytvoříme skupiny

```
else
{
$db->SQL_dotaz("SELECT ID_NOT, MIN(FUZZY) FROM fuzzy WHERE fuzzy >
0 AND ID_L LIKE '$slcd' GROUP BY ID_NOT");
while ($db->fetch_row())
{
$idnotebooku = $db->result("ID_NOT");
$fuzzynotebooku = $db->result("MIN(FUZZY)");
mysql_query("INSERT INTO nejhorsivlastnosti (ID_NO,MINFUZ)
VALUES('" . $idnotebooku . "','" . $fuzzynotebooku . "')");
} // uložíme si maximální hodnotu ohodnocení v předchozí nalezené skupině
$max = mysql_result(mysql_query("SELECT MAX(MINFUZ) FROM
nejhorsivlastnosti"), 0);
```

- najdeme dané ohodnocení v tabulce nejhorších vlastností

```
$db->SQL_dotaz("SELECT ID_NO FROM nejhorsivlastnosti WHERE
MINFUZ='$max'");
while ($db->fetch_row())
{
$idsearch = $db->result("ID_NO");
mysql_query("INSERT INTO pomoc (LING,NOTE)
VALUES('" . $slcd . "','" . $idsearch . "')") }
```

- jakmile máme předchozí kroky, provedeme pro všechny parametry ohodnocení notebooků a ty které dosáhly největšího ohodnocení, se vypíší

```
$db->SQL_dotaz("SELECT NOTEB FROM pomoc WHERE LING = '$slcd'");  
while ($db->fetch_row())  
{  
    $noteb = $db->result("NOTEB");  
    $pocet = $hodmax+1;  
    mysql_query("UPDATE pomoc SET POCET='$pocet' WHERE NOTEB = '$noteb'");  
}
```


5 E-SHOP S FUZZY EXPERTNÍM SYSTÉMEM

Internet, který v době psaní tohoto textu (2009) slaví 20-té narozeniny své existence, již není jen souhrnem hypertextových odkazů a jednoduché grafiky. Za svojí existence se několikrát zrychlilo připojení k internetu. Vznikla spousta robustních aplikací pro webové aplikace, např. redakční systémy, informační systémy, e-shopy, v dnešní době hodně oblíbené internetové komunikátory a tzv. sociální sítě (např. facebook.com, last.fm, myspace.com). Internet láká stále více uživatelů, kteří jej využívají k práci nebo ke studiu, např. k přístupu k firemnímu, případně školnímu informačnímu systému nebo jen k zjištění aktuálních informací, prohlédnutí videí a obrázků nebo k vyhledání informací. Se zvyšujícím počtem uživatelů začal být internet výborným místem pro obchodníky, kteří tak mohou nabízet své zboží nejen ve svých klasických obchodech, ale i na internetu pomocí internetových aplikací, díky kterým mohou oslovit mnohem více klientů [8].

Internetový obchod slouží k procházení katalogu zboží, ve většině případů v nějakých kategoriích a s detailním popisem zboží. Většina e-shopů nabízí vyhledávání zboží na základě požadavků uživatele. V našem případě k vyhledávání neurčitých údajů využijeme expertní znalostní systém, který jsme si navrhli v předchozí části výkladu. Tento systém nabízí rovněž možnost objednání, dopravy a platby za zboží. Sofistikovanější aplikace nabízí propojení e-shopu s účetní agendou, který obchod využívá, zobrazování počtu kusů na skladě, diskuse o zboží a platby pomocí platební karty přes internet. E-shop je navržen tak, aby administrátor aplikace měl co nejméně práce a o vyřizování objednávek a doplňování zboží se mohli starat třeba zaměstnanci obchodu [8].

5.1 Návrh tabulek

Pro zopakování připomeňme, že již v databázi máme k dispozici tabulku s konkrétními údaji notebooků, které nám poslouží k navržení aplikace katalogu. Dále máme k dispozici další dvě tabulky pro expertní systém, které jsme již definovali v kapitole 4. Nyní doplníme databázi o další tabulky důležité pro správný návrh e-shopu a jeho doplňkových funkcí. Základními požadavky našeho e-shopu byly stanoveny především následující funkce:

- ✓ Administrační systém,
- ✓ katalog zboží,
- ✓ košík,
- ✓ systém pro objednání zboží,
- ✓ systém expedice zboží,
- ✓ skladovací systém,
- ✓ diskuse

5.1.1 Tabulka administračního systému

Administrační systém slouží k autentizaci uživatele na základě jeho uživatelského jména a hesla. Jde o základní identifikaci uživatele na internetu. Žádné systémy na internetu se bez této funkce neobejdou. Následující tabulka uchovává přihlašovací údaje uživatele, korespondenční údaje pro odeslání zásilky, kontaktní údaje a úroveň práv, které má k dispozici uživatel.

Tab. 7 user.

Název sloupce	Typ sloupce	Popis
ID_U	int(10)	Slouží k identifikaci uživatele
USER	varchar(35)	Uživatelské jméno
HESLO	varchar(45)	Heslo uživatele
JMENO	varchar(35)	Jméno uživatele
PRIJMENI	varchar(35)	Příjmení uživatele
ULICE	varchar(40)	Ulice uživatele
MĚSTO	varchar(35)	Město uživatele
PSC	varchar(10)	PSČ uživatele
MAIL	varchar(50)	E-mail uživatele
MOBIL	varchar(20)	Telefonní kontakt uživatele
AUTH	varchar(1)	Nastavení práv uživatele
STAV	varchar(1)	Informace, zda účet byl již aktivován pomocí mailu
REG	varchar(40)	Kód pro aktivování účtu
NMAIL	varchar(35)	Nový e-mail, který se nastaví po potvrzení mailu.

Typy sloupců byly podrobně popsány v kapitole 4, v případě nejasností tedy odkazují na str. 33.

5.1.2 Virtuální nákupní košík

Je základní vlastností každého e-shopu. Lze si ho představit jako klasický nákupní košík. Virtuální nákupní košík tedy slouží k stejné činnosti jako klasický nákupní košík v supermarketu. A to k vkládání vybraného zboží do košíku a následný přechod k dokončení nákupu a objednání zboží, anebo odstranění položek z košíku. To znamená, že košík slouží jako část aplikace, která provádí zákazníka mezi katalogem zboží a samotnou objednávkou zboží.

Tab. 8 kosik.

Název sloupce	Typ sloupce	Popis
ID_K	int(10)	Identifikace košíku
DATUM	datetime	Datum přidání do košíku
ID_UZ	int(10)	Identifikace uživatele nákupu
ID_ZBOZ	int(10)	Identifikace přidaného zboží do košíku
KUSU	int(4)	Počet kusů
CENAN	varchar(7)	Celková cena

5.1.3 Tabulka systému k objednání zboží

Nyní, když máme vybráno zboží, můžeme přejít k jeho objednání. Ve většině případů si zde uživatel volí, jakým způsobem si přeje zboží převzít, tj. osobně na pobočce, poštou, PPL nebo kurýrem a dále, jakým způsobem zaplatí za zboží, tj. neprodleně platební kartou, příkazem k úhradě, hotově při převzetí zboží nebo sjednáním úvěru. Po odeslání objednávky se uloží do následující tabulky. Všechny objednávky má k dispozici prodejce po svém přihlášení do aplikace, aby byl schopný připravit zboží k expedici k zákazníkovi.

Tab. 9 objednávka.

Název sloupce	Typ sloupce	Popis
ID_O	int(10)	Identifikace objednávky
ID_KLIENT	int(10)	Identifikace zákazníka
ID_ZBOZI	int(10)	Identifikace zboží
DATE	datetime	Datum objednávky
VS	varchar(15)	Variabilní symbol objednávky
DODANI	varchar(5)	Dodání zboží, dopravou nebo osobně
ZPUSOB	varchar(5)	Způsob dodání, tj. na které pobočce nebo jakým způsobem dopravy
PLATBA	varchar(5)	Jakým způsobem bude provedena platba
KUS	int(10)	Počet kusů zboží
CENAVKOSIKU	varchar(10)	Cena zboží v košíku
CENA_CELKEM	varchar(12)	Celková cena zboží
POZN	varchar(30)	Doplňující informace k platbě
VYRIZENI	int(2)	Stav vyřízení objednávky
DVYRIZENI	datetime	Datum poslední změny objednávky

5.1.4 Skladovací systém

Při návrhu aplikace byl požadavek, aby byl zákazník co nejlépe informovaný o zboží, a proto byl do aplikace implementován skladovací systém, který nás informuje, kolik notebooků je na skladech. Na základě této informace se odvíjí i doba dodání notebooku zákazníkovi. Protože je jasné, že pokud není notebook ani na jednom skladě našeho obchodního řetězce, bude se muset objednat od dodavatele, což samozřejmě trvá déle.

Tab. 10 sklad.

Název sloupce	Typ sloupce	Popis
ID_S	int(10)	Identifikace notebooku
EXTERNI	varchar(10)	Externí sklad
PRAHA	varchar(10)	Sklad v Praze
BRNO	varchar(10)	Sklad v Brně
OLOMOUC	varchar(10)	Sklad v Olomouci
OSTRAVA	varchar(10)	Sklad v Ostravě

5.1.5 Tabulka dopravních informací

Následující tabulka je v naší aplikaci v podstatě jen doplňující částí našeho objednávacího systému, u kterého odesíláme zboží zákazníkovi přes třetí stranu, tj. externí firmy zabývající se dopravou balíků. Tyto společnosti však mohou postupem času změnit své služby, např. zrychlit dopravu nebo zdražit poskytování svých služeb. Následující tabulka zajišťuje editaci těchto změn. Bez nutnosti zásahu do zdrojového kódu aplikace.

Tab. 11 doprava.

Název sloupce	Typ sloupce	Popis
ID_D	int(10)	Identifikace dopravy
NAZEVD	varchar(35)	Název dopravy zboží (pošta, PPL atd.)
CENAD	varchar(10)	Cena za zboží
DOBA	varchar(35)	Doba dopravy

5.1.6 Tabulka diskuse k notebookům

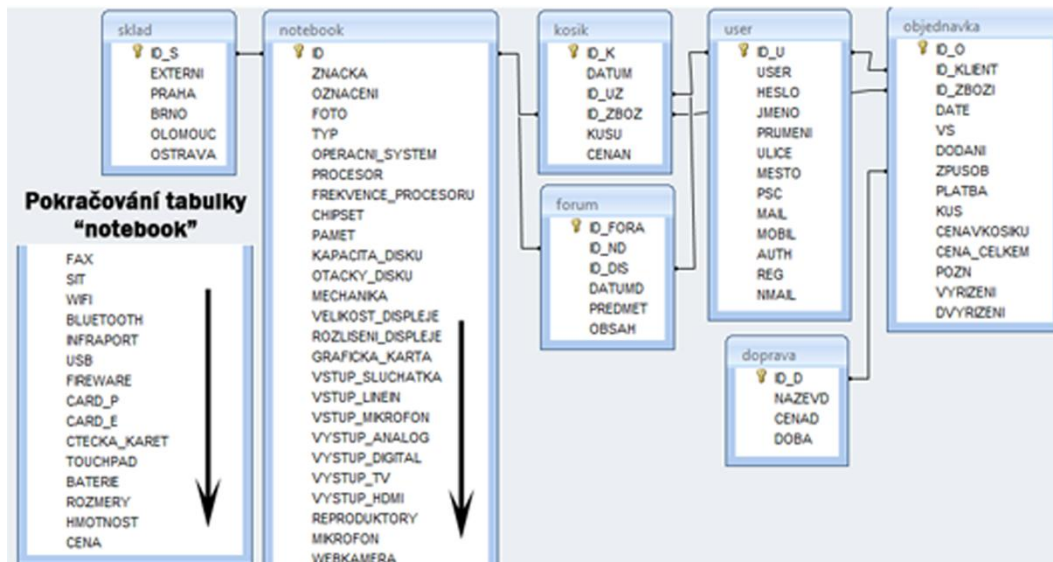
Poslední tabulka slouží jako prostředek komunikace mezi ostatními zákazníky a prodejci obchodu, tj. pomocí diskuse mohou zákazníci psát zkušenosti s daným notebookem, případně se zeptat na nějakou doplňující otázku obsluhy, pokud jim není vše jasné.

Tab. 12 forum.

Název sloupce	Typ sloupce	Popis
ID_FORA	int(10)	Identifikace fóra
ID_NO	int(10)	Identifikace notebooku
ID_DIS	int(10)	Identifikace diskuse
DATUMD	datetime	Datum napsání diskuse
PŘEDMĚT	varchar(128)	Předmět diskuse
OBSAH	varchar(1000)	Obsluha diskuse

5.1.7 Relace

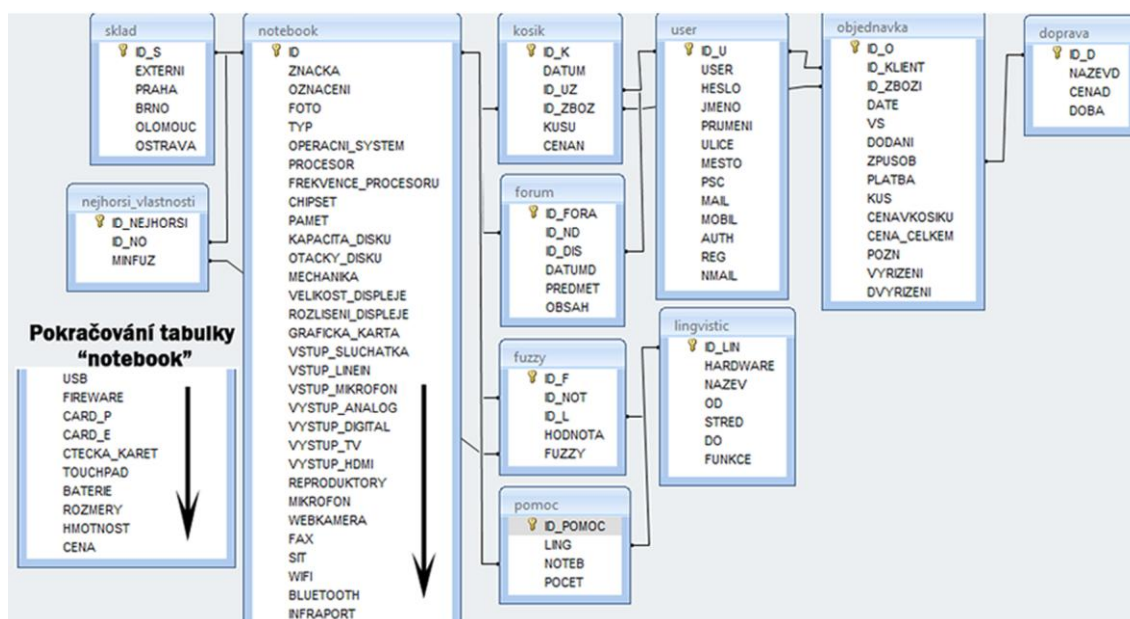
Nyní, když máme již popsané všechny tabulky, můžeme přistoupit k relaci databáze. Následující Obr. 22 zobrazuje relaci mezi výše navrženými tabulkami našeho e-shopu. Na obrázku je vidět vazby všech tabulek e-shopu, a také propojení tabulky objednávka s tabulkou notebook, která je vedena přes tabulku košík. Při návrhu datových struktur jsme vycházeli z metodologie normálních forem relací s cílem dosáhnout u všech tabulek minimálně třetí normální formy [6].



Obr. 22 Relace tabulek e-shopu.

5.2 Relace všech tabulek v databázi naší aplikace

Nyní uvedeme pro větší názornost kompletní relaci všech tabulek v databázi, která zajišťuje funkčnost naší aplikace. Ve větším rozlišení je k dispozici kompletní relace tabulek v databázi v příloze na Obr. 46.



Obr. 23 Relace všech tabulek v databázi naší aplikace.

5.3 Administrační systém

Pro co nejjednodušší autentizaci uživatelů se používá administrační systém. Kromě přihlášení do systému nabízí administrační systém úrovně práv, tj. jaké položky v menu aplikace se zobrazí po přihlášení uživatele [10]. Mezi jeho základní funkce patří

- registrace,
- přihlášení a odhlášení,
- nastavení účtu,
- přidělování práv uživatelům aplikace.

Z hlediska implementace administračního systému je důležité zajistit šifrování citlivých informací získaných od uživatele tak, aby tato data nemohla být nikým zneužitelná, a to ani administrátorem aplikace. Mezi citlivé informace především patří uživatelské heslo, kterým se přihlašujeme do systému. Heslo je v aplikaci hasíváno a ke každému kódu je přiřazen řetězec určitých znaků, které zajišťují ještě větší bezpečnost uživatelských hesel v naší databázi.

Jak vyplývá z tabulky User, kromě přihlašovacích údajů uživatele a nastavení práv jsou uloženy v tabulce i korespondenční údaje sloužící k expedici objednaného zboží a kontaktních údajů, kde především na e-mail odesílá tento systém údaje o objednaném zboží v e-shopu a informace o průběhu objednávky. Tyto výše uvedené informace o uživateli není potřeba zadávat již při registraci do systému, ale je nutné je vyplnit při první objednávce zboží, jinak aplikace nepovolí odeslání objednávky k zpracování.



Obr. 24 Nepřihlášený uživatel.



Obr. 25 Přihlášený uživatel.

Na Obr. 24 jsou vidět možnosti pro přihlášení a registraci. Po přihlášení se změní nepřihlášen na jméno uživatele a do hranaté závorky se napíše práva, jaké byly uživateli administrátorem aplikace přiděleny. Tlačítka se změní na Odhlásit se a Nastavení (viz Obr. 25).

Implementace přihlášení je řešena pomocí sessions, tj. při přihlášení uživatele si do sessions uložíme id uživatele a jeho práva, které si zjistíme v databázi. Dokud nezrušíme sessions odhlášením z aplikace nebo smazáním sessions, v internetovém prohlížeči aplikace bude stále přihlášen. Vypršení sessions je automaticky nastaveno na 30 minut, tj. pokud nebude uživatel po tuto dobu pracovat s aplikací, tak bude z aplikace automaticky odhlášen.

Pomocí práv, které si při přihlášení ukládáme do sessions, se testuje, zda stránky aplikace mají být pro daného uživatele přístupné nebo ne. Pokud danou stránku nemá uživatel přístupnou, tak se odkaz na tuto stránku nezobrazí v menu aplikace. V případě, že by se uživatel dozvěděl nějakým způsobem odkaz na stránku, kterou nemá k dispozici, proběhne kontrola přístupu a vypíše se mu příslušná informace.

5.3.1 Registrace a nastavení účtů

Abychom se mohli přihlásit do naší aplikace, musíme se nejprve zaregistrovat. K tomu účelu stačí vyplnit registrační formulář. Pro registraci je nutno splnit dva nejdůležitější požadavky a to unikátní:

- uživatelské jméno,
- e-mail uživatele,

v databázi aplikace. E-mail se používá především k aktivaci účtu. Dokud účet neaktivujeme pomocí aktivačního e-mailu, nebude možné vytvořený účet používat, dále slouží k potvrzení změny e-mailu v aplikaci, anebo se využívá pro obnovu zapomenutého hesla. Jako autorizační kód se používá vygenerovaný URL odkaz doplněný o hash, na základě kterého se provede požadavek.

K zaregistrování si musíme nejprve zvolit nějaké unikátní uživatelské jméno a pomocí tlačítka Ověřit zjistíme, jestli je volné. Pokud ano, můžeme pokračovat v registraci (viz. Obr. 26).

Uživatelské jméno:

Uživatelské jméno je volné

Heslo:

Kontrola hesla:

E-mail:

Osobní informace:

Jméno:

Příjmení:

Ulice:

Město:

PSČ:

Mobilní telefon:

Souhlasím se zpracováním osobních údajů: ☐

Obr. 26 Formulář k registraci do aplikace.

Jakmile vyplníme výše uvedený formulář a odešleme, tak v tom okamžiku se automaticky odešle na uživatelem zvolený e-mail aktivační požadavek nově vytvořeného účtu (viz Obr. 27). Jakmile použijeme odkaz v e-mailu, dojde k aktivaci účtu.

Potvrzení registrace

Od: **Admin** (rott@czfinance.cz)
Odesláno: 19. května 2009 23:40:03
Komu: majkl.knight@hotmail.com

Dobrý den

Děkujeme vám za registraci na našich stránkách Fuzzy shopu notebooku.
Pro používání vašeho účtu musíte nejprve potvrdit registraci kliknutím na odkaz: <http://fuzzy.czfinance.cz/login.php?potvrz=822a8f2c644e544b0e24cd9e870dbec15b616a4d>.

Vaše přihlašovací údaje:

Uživatelské jméno: pepa
Heslo: pepek1

[Fuzzy shop notebooku](#)

Pokud jste se na našem portálu neregistroval, berte tento e-mail za bezpředmětný.

Obr. 27 Ukázka potvrzovacího e-mailu.

Na Obr. 28 je ukázka možnosti nastavení uživatelského účtu, v první části je vidět možnost změny hesla, dále nastavení nového e-mailu a v poslední části je možnost změny osobních údajů.

Nastavení nového hesla:

Původní heslo:
Nové heslo:
Kontrola hesla:

Nastavení nového e-mailu:

Původní e-mail:
Nový e-mail:

Nastavení osobních údajů:

Jméno:
Příjmení:
Ulice:
Město:
PSČ:
Mobilní telefon:

Obr. 28 Ukázka možnosti nastavení uživatelského účtu v aplikaci.

Implementace registrace a nastavení účtu se provádí pomocí informací získaných z formuláře a následné uložení do databáze pomocí jazyka SQL. Pro odesílání e-mailů

se využívá funkce php pro odeslání automatického e-mailu. Hashování hesla a aktivačního kódu se provádí také pomocí funkce v php pro hashování citlivých údajů [3].

5.3.2 Uživatelská práva a jejich přidělování

S autorizací uživatelů úzce souvisí i přidělování práv uživatelům. Přidělováním těchto práv definujeme, kdo má jaká práva, např. zákazník, který si u nás koupí zboží, zcela jistě nesmí mít přístup do editace a modelování expertního znalostního systému nebo do administrace vyřizování objednávek. Naším požadavkem je zajistit, aby k expertnímu znalostnímu systému měli přístup jen lidé s požadovanými znalostmi. V naší aplikaci byly navrženy tyto kategorie práv:

- **Nepřihlášený** – náhodný návštěvník e-shopu. Můžeme procházet katalog, novinky, přehled, jestli je v některém skladě zboží skladem a vyhledávat i notebooky dle svých požadavků. V případě, že zvolíme košík nebo zobrazení diskuse, tak tohoto uživatele aplikace požádá o přihlášení nebo registraci.
- **User** – uživatel získá po přihlášení nebo registraci a aktivaci nového účtu toto právo automaticky přiděleno od aplikace. Můžeme provádět vše, co nepřihlášený uživatel a navíc můžeme přispívat do diskuse, vidět přesné počty zboží v jednotlivých skladech, provádět objednávky zboží a nahlížet na průběh a historii svých objednávek.
- **Redaktor** – následující právo musí uživateli přiřadit administrátor aplikace. Můžeme provádět vše, co u předchozích oprávnění a navíc máme práva na vyřizování objednávek, doplňování a editaci notebooků, možnost doplňování zásob ve skladech a souhrn všech zpráv ve fóru pro případné odpovědi uživatelům. Toto právo je ušito na míru pro prodavače nebo osoby, které se budou starat o objednávky.
- **Admin** - následující právo musí přidělit superadmin (viz dále). Tento uživatel může vše, co uživatelé s výše popsanými oprávněními. Navíc má práva k přidělování práv podřazených skupin, tj. redaktor a user, dále má přístup k editaci a modelování lingvistických proměnných a generování fuzzy ohodnocení notebooků, v neposlední řadě má přístup k editaci cen a doby dopravy. To oprávnění je určeno pro znalostní inženýry (experty).
- **SuperAdmin** – tento účet je defaultně vytvořen při nainstalování aplikace administrátorem a je jen jeden. Jde o nejvyšší stupeň administrace. Má všechna práva předchozích tříd uživatelů a navíc má právo pro nastavení práv adminům. Následující právo SuperAdmina má pouze programátor nebo majitel aplikace.

Všechna výše definovaná práva fungují na principu, že mohou vždy měnit pouze práva svých podřízených, ale uživatelé, kteří mají stejná práva nebo vyšší, již uživatel nevidí. V praxi to znamená, že SuperAdmin má práva na změnu práv všem uživatelům a dále Admin má práva jen na změnu práv redaktorů a klasických uživatelů. Redaktor již nemůže nikomu práva měnit.

Následující koncepce práv byla navržena z důvodů, aby administrátor nemusel složitě vytvářet účet pro své zaměstnance, a to ať již se jedná o lidi, kteří se budou starat o vyřizování objednávek, doplňování nových notebooků do databáze, doplňování zboží ve skladech, nebo se bude jednat o znalostní inženýry, kteří se budou starat o správnou funkci znalostního expertního systému. Výše zmíněné uživatelé se požádá, aby se přihlásili do systému a po přihlášení a aktivaci mu administrátor aplikace přidělí příslušná práva. Obr. 29 znázorňuje ukázkou nastavení práv tak, jak ho má k dispozici uživatel s právy

SuperAdmina. V případě Admina je tato administrace stejná, jen bychom neviděli uživatele, kteří mají práva Admina.

<u>Nastavení práv</u>			
Jméno a příjmení:	Uživatelské jméno:	Stav:	Práva:
Petr Morávek	<u>majkl.knight</u>	A	Admin ▾ <input type="button" value="Nastavit"/>
Pavel Morávek	<u>majkl</u>	A	Redaktor ▾ <input type="button" value="Nastavit"/>
	<u>anetka</u>	A	User ▾ <input type="button" value="Nastavit"/>
Jan Novák	<u>pokus</u>	A	User ▾ <input type="button" value="Nastavit"/>
Josef Novák	<u>pepa</u>	A	User ▾ <input type="button" value="Nastavit"/>

Obr. 29 Ukázka administrace práv uživatele SuperAdmin.

Nyní si popíšeme sloupce v tabulce pro nastavení práv, kterou můžeme vidět na Obr. 29:

- **Jméno a příjmení** – hodnoty, které jsou prázdné, znamenají, že daný uživatel nevyplnil své osobní údaje => nemůže objednávat.
- **Uživatelské jméno** – po kliknutí se zobrazí podrobné informace o uživateli.
- **Stav** – A = aktivní, N = neaktivní.
- **Práva** – editace a nastavení práv pro daného uživatele.

5.4 Katalog notebooků

Abychom mohli prodávat zboží na internetu, musíme nějakým sofistikovaným způsobem naše zboží na internetu prezentovat. K tomuto účelu se používá katalog zboží. Skládá se ze dvou základních kategorií, a to především katalog pro uživatele, kde využíváme lingvistických proměnných navržených v našem expertním systému, a dále pak katalog pro odborníky, který je podobný všem katalogům známým z různých e-shopů na internetu. Oba tyto katalogy si podrobně popíšeme v další části textu. Se samotným katalogem jsme se již setkali v kapitole 4.4. V každém katalogu jsou další tři podkategorie, jde především o tři nejzákladnější požadavky na notebook, které jsou pro uživatele ve většině případů nejdůležitější. Jde o značku, velikost LCD a cenu.

Na jedné stránce katalogu se zobrazí přehled 10 notebooků a pro přechod na další stránku nám stačí zvolit číslo stránky, kterou požadujeme nebo zvolit šipku. Výběry stránek jsou umístěny na pravé straně stránky v dolní i horní části tak, aby byly neustále k dispozici.


Přehled stránky se skládá z obrázku notebooků a vedle každého obrázku je uveden základní popis notebooků včetně ceny za zboží, informace o množství zboží ve skladech našeho obchodu a tlačítka pro vložení do košíku viz Obr. 30.

Když zvolíme pomocí myši název notebooků nebo obrázek notebooků, v přehledu všech notebooků se nám zobrazí podrobný popis veškeré výbavy notebooků. V případě, že jsme přihlášení, tak se navíc zobrazí, kolik notebooků je ve skladech obchodního řetězce. Podrobný popis obsahuje všechny důležité informace o daném notebooku, viz Obr. 31.

Pod podrobnostmi o notebooku se nachází diskuse. Na stejné stránce se zobrazí dvě poslední zprávy a v případě, že si požadujeme přečíst celou diskusi, máme možnost jí zobrazit pomocí tlačítka pro vstup do diskuse.

	<p><u>APPLE MacBook Pro</u></p> <p>Intel Core 2 Duo T9300 2.5 GHz, paměť DDR2 4 GB, pevný disk 320 GB (5400 ot./min), velikost displeje 15.4", rozlišení displeje 1920x1200, grafická karta NVIDIA GeForce 8600M, síťová karta, wifi karta, bluetooth, SD/MMC/MS/MS Pro, 2.99 kg</p> <p>Cena: 64 258,- Kč</p>
	<p><u>APPLE MacBook Air</u></p> <p>Intel Core 2 Duo 1.86 GHz, paměť DDR2 2 GB, pevný disk 128 GB (4200 ot./min), velikost displeje 13.3", rozlišení displeje 1280 x 800, grafická karta NVIDIA GeForce 9400M, , wifi karta, bluetooth, infraport, 1.36 kg</p> <p>Cena: 69 851,- Kč</p>

Obr. 30 Ukázka přehledu katalogu notebooků.

<p><u>Značka APPLE MacBook Pro</u> 64 258,- Kč</p>	
<p><i>Sklad: Externí: 4ks, Praha: 1ks, Brno: 1ks, Olomouc: 0ks, Ostrava: 1ks</i></p>	
	<p><u>Výkon:</u></p> <p>Procesor: Intel Core 2 Duo T9300 2.5GHz Chipset: Intel 965PM Express Velikost paměti: 4GB</p> <p><u>Displej:</u></p> <p>Velikost displeje: 15.4 Rozlišení displeje: 1920x1200 Grafická karta: NVIDIA GeForce 8600M</p>
<p><u>Systém:</u></p> <p>Operační systém: Mac OS X v10.5 Leopard</p>	
<p><u>Vstupy:</u></p> <p>Sluchátka: ano Line-in: ano Mikrofon: ano</p>	<p><u>Disky a mechaniky:</u></p> <p>Velikost disku: 320 GB Otáčky disku: 5400 ot./min Mechanika: DVD±RW SuperDrive</p>
<p><u>Technické parametry:</u></p> <p>Baterka: 6 článková Rozměry: 392 x 265 x 25,9 mm Hmotnost: 2.99 kg</p>	
<p><u>Integrované:</u></p> <p>Reproduktory: ano Mikrofon: ano Webkamera: Mpx Fireware: ano PC Card: ne Express Card: ano Touchpad: ano Čtečka karet: SD/MMC/MS/MS Pro</p>	<p><u>Komunikace:</u></p> <p>Síť: ano Wi-fi: ano Bluetooth: ano Infraport: ne Fax modem: ne USB: 3 x</p>

Obr. 31 Podrobný popis notebooků v katalogu.

Implementace katalogu je řešena načítáním informací pomocí SQL příkazů, následné zobrazení v daném formátu se řeší cyklem definovaným v PHP. Detail stránky se vyhledá v databázi pomocí id notebooku, které odešleme aplikaci při zvolení myší obrázku nebo nadpisu v přehledu katalogu notebooků.

5.4.1 Vložení nového, editace a smazání notebooků

Přístup k možnosti vložení notebooků mají všichni uživatelé, kteří mají práva redaktora nebo vyšší. K samotnému vložení slouží formulář, který obsahuje všechny údaje o daném notebooku. Po vyplnění formuláře stačí zvolit tlačítko Uložit a notebook se zavede do systému (viz. Obr. 32). U podrobných informací o notebookech je k dispozici tlačítko, pomocí něhož můžeme vstoupit k editačnímu formuláři (viz. Obr. 32). Je úplně stejný jako ten, který používáme pro vložení nového notebooku. Součástí těchto formulářů je i další formulář, který slouží k zadání počtu notebooků ve skladech již při vkládání notebooků. Zvolením tlačítka smazat se zobrazí potvrzovací otázka, jestli opravdu si přejeme daný notebook smazat. Přístup k tomuto tlačítku mají redaktoři nebo vyšší, stejně jako u editací a vložení. Protože se jedná ve všech případech o stejné formuláře, je pro ilustraci uveden pouze formulář pro editaci, viz Obr. 32.

Obr. 32 Formulář k vložení nového a editaci notebooku.

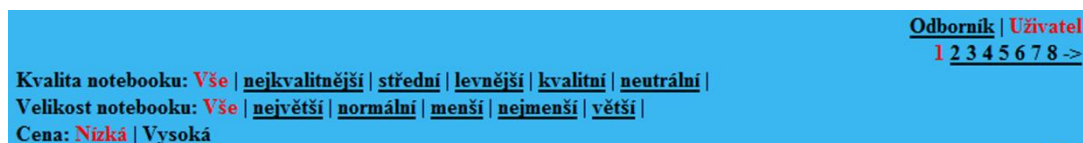
Implementace se provádí pomocí SQL příkazů pro editaci nebo smazání řádku. Editace i smazání notebooků je spojena s fuzzy expertním systémem, tj. pokud smažeme nějaký notebook, tak tento notebook si po sobě tzv. „uklidí“ i v tabulce „fuzzy“, která slouží k ohodnocení parametrů. Stejně tak, pokud změníme nějaký parametr notebooků při editaci, dojde automaticky i k přepočítání a nahrazení ohodnocení v případě, že změněný parametr je ohodnocujícím parametrem.

5.4.2 Katalog pro uživatele

Katalog pro uživatele se defaultně zobrazuje při výběru katalogu notebooků. Uživatelský katalog využívá lingvistických proměnných v kategoriích notebooků na základě, kterých

zpřesňujeme požadavky na výběr notebooků v katalogu. Tento katalog nabízí výběr notebooků pomocí výrazů přirozeného jazyka, které nahrazují přesné informace obsažené v katalogu pro odborníky.

Na Obr. 31 vidíme kvalitu, velikost a cenu notebooků. Vedle toho jsou vidět lingvistické proměnné, které jsme modelovali v expertním znalostním systému. Na pravé straně je možnost na přechod do katalogu pro experty a dále možnosti přechodu na další stránku katalogu.

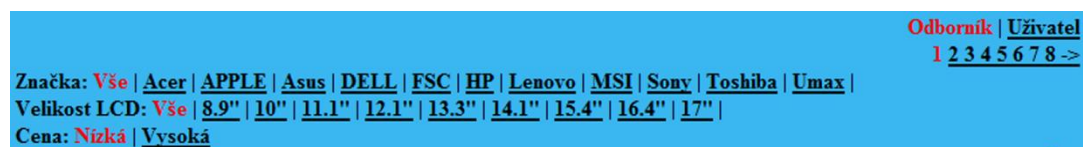


Obr. 33 Kategorie v katalogu pro uživatele.

5.4.3 Katalog pro odborníky

Jde o alternativu ke katalogu pro uživatele. Při návrhu aplikace byl tento katalog navržen jako první, protože bylo potřeba nejprve naplnit databázi konkrétními daty notebooků, aby měl expertní systém co ohodnocovat. Dále bylo třeba vytvořit katalog, kde by byly kategorie notebooků popsány lingvistickými proměnnými, a tak vznikl katalog pro uživatele.

Na Obr. 34 je vidět značka, velikost LCD a vedle toho jsou konkrétní informace o značkách a velikostech LCD. Takto navržené kategorie můžeme znát z různých e-shopů na internetu.



Obr. 34 Kategorie v katalogu pro odborníky.

5.5 Systém k objednání zboží

Nejdůležitější částí elektronického obchodu je objednávání zboží. Součástí systému k objednání zboží je možnost dopravy zboží na adresu zákazníka, anebo vyzvednutí zboží na prodejně. V dnešní době je velmi moderní platba za zboží pomocí platební karty nebo vyřízením úvěru. Na všechny možnosti bylo při vývoji aplikace pamatováno.

Uživatel pro to, aby si mohl objednat zboží, musí být přihlášen do systému. S implementací též souvisí tabulky, které již byly definovány v předchozím textu. V našem případě s objednacím systémem souvisí především tabulka kosík a objednávka, kde první tabulka slouží k ukládání informací o tom, co si daný uživatel chce objednat, druhá slouží k vytvoření objednávky a současně k zobrazení informací obsluze obchodu, která zboží z objednávky expeduje.

Při výběru košíku se pomocí sessions přesunou data o uživateli i o objednaném zboží do tabulky košíku. Tlačítka košíku jsme si mohli všimnout na Obr. 28 nebo Obr. 29, kde se vždy nachází vpravo nahoře. Současně při zvolení košíku se do sessions uloží i aktuální stránka, na které se uživatel nachází a jakmile se provede kód, který má za úkol naplnit tabulku daty získanými od uživatele, ze sessions se načte znovu stránka, na které byla operace výběru zahájena a uživatel může pokračovat ve výběru dalšího zboží. V pravé

liště aplikace košíku je zobrazena informace o počtu vybraných položek a celkové ceně za zboží. Tyto údaje se zjistí jednoduchým dotazem, který určí počet řádků v tabulce košík pro daného uživatele a následně sečte položky cena ve všech řádcích. V pravé liště je navíc i odkaz pro vstup do košíku, viz Obr. 35.



Obr. 35 Ukázka informací o objednávce.

5.5.1 Košík

Vybereme-li košík v pravé liště naší aplikace, dojde k vyhledání všech informací o zboží, které si daný uživatel vybral. Identifikace uživatele se provede na základě uložení jeho id v sessions tak, jak bylo popsáno v kapitole 5.2. Na základě tohoto id notebooku jsme pak schopni vyhledat všechny informace o zboží vloženém do tabulky košík. Po vyhledání všech informací o objednaném zboží se tyto informace zobrazí na stránce aplikace, viz Obr. 36.

Váš nákupní košík obsahuje:				
Obrázek	Podrobnosti	Kusů	Cena celkem	Del
	APPLE MacBook Intel Core 2 Duo 2.4 GHz, paměť DDR2 2GB, pevný disk 250 GB velikost displeje 13.3", rozlišení displeje 1440x900, grafická karta Integrovaná NVIDIA GeForce 9400M síťová karta, wifi karta, bluetooth, SD, MMC, MS, MS PRO, xD, 2.04 kg Datum vložení do košíku: 20.05.2009 00:40:03	1 ks	45 456,-	X
	Sony VAIO CR41Z/R Intel® Core2 Duo™ T8300 2.4 GHz, paměť DDR2 4 GB, pevný disk 250 GB velikost displeje 14.1", rozlišení displeje 1280 x 800, grafická karta ATI® Mobility Radeon™ X2300 síťová karta, wifi karta, bluetooth, SD, MMC, MS, MS-Pro, mini SD, 2.5 kg Datum vložení do košíku: 20.05.2009 00:40:29	2 ks	51 160,-	X
Celková cena vašeho nákupu: 96 616,- Kč				

Obr. 36 Ukázka obsahu košíku.

Na obrázku je vidět, že opět se využilo základního katalogu podrobně popsaného v předchozí kapitole 5.3, který je doplněn o další možnosti nastavení.

První možností nastavení je počet kusů. Pokud tento formulář odešleme, tak kód provede opravu údaje ve sloupci počtu kusů pomocí id košíku. Následuje cena za zboží, která se vypočítá jako počet kusů vynásobeným cenou za kus, kterou jsme si zjistili

v katalogu zboží. Na konci je vidět křížek, který slouží k odstranění zboží z košíku, které způsobí odstranění řádku objednávky na základě id košíku z tabulky košík.

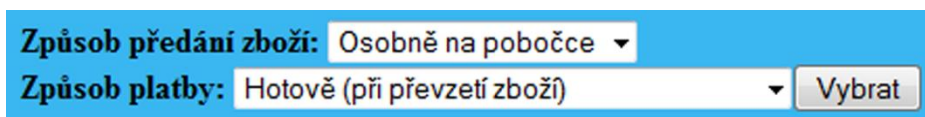
Na konci této stránky máme k dispozici celkovou cenu za zboží a následně tlačítka Odebrat vše z košíku a Přejít k výběru dopravy a platbě. Po zvolení tlačka Odebrat vše z košíku dojde k odstranění všech řádků, které ve sloupci id_uz obsahuje id uživatele, které je uloženo v sessions. V případě, že jsme spokojeni s objednávkou, můžeme přistoupit k dokončení, v opačném případě se můžeme vrátit do katalogu a pokračovat ve výběru zboží.

5.5.2 Dokončení objednávky

Objedávka se skládá z několika otázek, pomocí kterých dáme informaci prodejci zboží, jestli má zboží poslat na naši korespondenční adresu nebo si ji vyzvedneme v některé z jeho prodejen. Další otázky souvisí s informací, jakým způsobem si přejeme zaplatit za zboží. Po vyplnění všech těchto údajů dojde k uložení aktuálního data, zboží a výše zadaných informací do tabulky objednávka, tj. tabulka, ke které mají přístup zaměstnanci a administrátoři aplikace.

Výše definované informace jsou implementovány pomocí několika po sobě následujících kroků, které dovedou uživatele k potvrzení objednávky. Informace získané v těchto krocích jsou postupně ukládány do sessions a jakmile zadáme požadavek k vložení dat do databáze – dojde k přesunutí dat z sessions. Jakmile dokončíme objednávku, aplikace odešle zákazníkovi potvrzovací e-mail se seznamem zboží, které si objednal.

Jakmile zvolíme tlačítko Přejít k výběru dopravy a platbě v košíku, přesuneme se k dokončení aplikace, která v prvním kroku obsahuje informaci o dopravě a platbě za zboží, viz Obr. 37.



Obr. 37 Ukázka prvního kroku dokončení objednávky.

5.5.2.1 Doprava zboží

V prvním kroku máme na výběr dvě možnosti pro předání zboží, každá z těchto dvou kategorií má další čtyři podkategorie, které se generují podle zadaných informací z prvního kroku:

- Osobně na pobočce
 - Praha
 - Brno
 - Olomouc
 - Ostrava
- Dopravou
 - Pošta
 - PPL
 - Kurýr Brno
 - Kurýr Praha

V případě volby dopravy zboží se poslední dvě možnosti, tj. kurýr Brno a Praha, zobrazí v případě, že si zvolíme platbu platební kartou. Dále pokud si uživatel zvolí dopravu zboží pomocí kurýra, kontroluje se při odeslání zboží, jestli je uživatel opravdu z Brna nebo Prahy. V druhém případě, tj. převzetí na pobočce, si zákazník může vybrat město a tím i pobočku, kde si zboží vyzvedne.

V aplikaci je implementován skladovací systém, o který bude popsán v další části textu. Pokud uživatel požaduje, že si převezme zboží na prodejně, aplikace zkontroluje, jestli dané zboží je na skladě a pokud ano, neprodleně toto zboží odečte ze skladu a rezervuje pro uživatele, kterému pošle informaci, že do dvou hodin bude jeho zboží připraveno k převzetí na prodejně. V případě platby platební kartou a výběru dopravy kurýra se též kontroluje, jestli je zboží ve skladě daného města a v neposlední řadě dochází k odečítání zboží z externího skladu, pokud uživatel vybere dopravu poštou nebo PPL.

5.5.2.2 Platba za zboží

Možnosti platby se skládají ze čtyř základních variant, které souvisí s klasickými možnostmi platby, jaké se dnes nejčastěji používají:

- **Hotově (při převzetí zboží)** – v případě, že si uživatel vybere způsob dopravy osobně při převzetí zboží, zaplatí při návštěvě pobočky, v případě dopravy poštou nebo PPL zaplatí při převzetí balíku.
- **Bankovním převodem** – další možnost platby, kdy člověk zaplatí převodem a po připsání peněz na účet se mu buď připraví objednávka na prodejně nebo se mu zašle balík, který již nemusí při převzetí platit. Protože v dnešní době je bezhotovostní styk preferován, zákazník získá 2% slevu z celkové ceny.
- **Online platební kartou** – nejmodernější a nejrychlejší způsob platby, musíme mít ovšem u své banky povolenou platbu na internetu. Tento způsob platby je v západních státech nejvyužívanějším způsobem platby za zboží na internetu, u nás se ještě moc nerozšířil z důvodu nedůvěry a strachu o své peníze. Tento způsob platby má několik výhod, možnost rychlé dopravy po Brně nebo Praze, sleva 5% a zákazník navíc nemusí mít při převzetí zásilky u sebe peníze.
- **Úvěrem** – v případě, že uživatel nemá dostatek peněz k dispozici, je možnost si zvolit platbu pomocí úvěru, v tomto případě mu aplikace spočítá měsíční splátku, počet splátek a celkové navýšení, tzv. RPSN je stanoveno od 12%.

Při potvrzení objednaného zboží dojde zákazníkovi potvrzovací e-mail se seznamem zboží, které si objednal. V případě platby pomocí platební karty přes internet mu přijde ještě další e-mail s potvrzením o platbě za zboží.

Na závěr je nutno podotknout, že především platba pomocí platební karty je v našem případě pouze simulovaná. Takže není řešena žádná bezpečnost při vkládání platebních údajů z platební karty a formulář vytvořený k zadávání těchto údajů je tzv. „hluchý“, tj. data vložená do formuláře si nikam neukládá ani jiným způsobem nepracovává. V případě nasazení aplikace do ostrého provozu by se naše simulace nahradila přesměrováním na stránku banky, u které máme objednanou službu platby za zboží na internetu, protože platba za zboží je výhradně v režii banky a my jako správci internetového obchodu získáme pouze informaci od banky o tom, jestli platba proběhla nebo ne.

Způsob předání zboží:

Způsob platby:

Způsob dopravy zboží:

Informace o dodání zboží:

Vyřízení: Vaše objednávka bude odeslána do 24 hod.
 Platba: Online platební kartou
 Doprava: PPL. Zásilka k vám dorazí do 24 hod. Cena: 120,- Kč

Typ online platebních karet



Při použití online platební karty 5% sleva

Celková cena vašeho nákupu: 92 130,- Kč

Obr. 38 Ukázka dokončení objednávky a platby pomocí platební karty.

5.5.3 Vyřizování objednávek

Po dokončení objednávek nastává okamžik, kdy je potřeba vyřídit objednávku a předat zboží zákazníkovi. Samotné vyřízení objednávek se skládá ze dvou základních operací, kde jedna je zpětnou vazbou pro uživatele té druhé:

- Uživatel má možnost prohlížet stav svých objednávek.
- Zaměstnanci nebo administrátoři webu vidí seznam všech objednávek, které musí vyřídit a předat zboží zákazníkovi.

5.5.3.1 Moje objednávky

Každý uživatel, který se přihlásí do systému, má možnost se podívat na průběh svých objednávek a dále na historii objednávek. Tato část aplikace je zpětnou vazbou pro uživatele, aby měli přehled, co se děje s jejich objednaným zbožím. Uživatel dále obdrží z aplikace automaticky generovaný e-mail při jakékoli změně stavu objednaného zboží, který provede obsluha e-shopu.

Implementace je řešena načtením všech dat pomocí id uživatele získaného z sessions a následně se data rozdělí podle stavu objednávky. Do historie se přesunou všechny vyřízené objednávky a v režimu objednávek zůstanou všechny ostatní objednávky. Následně dojde k seřazení podle aktuálního data, viz Obr. 39.

Objednané zboží má vždy určitou barvou, kde každá barva má svůj specifický význam. Jejich význam je následující:

- ✓ **Zelená** – objednávka k převzetí v průběhu nebo vyřízená v historii objednávek.
- ✓ **Modrá** – objednávka se musí objednat z nějakého jiného dceřiného obchodu.

- ✓ **Červená** – objednávka se musí objednat od dodavatele nebo v historii se jedná o stornovanou objednávku.

Průběh objednávek						
Datum Hodina	Variabilní symbol	Předání zboží	Platba	Celk. kusů	Cena celkem	Datum Vyřízení
20.05.2009 00:49:38	8600993915	Dopravení pomocí PPL [120,- Kč, do 24 hod.]	Online platební kartou Typ: MasterCard	3	96 736,- Kč	20.05.2009 - 00:54:09 Objednáno od výrobce Zboží bude k dispozici do 3dní.
10.05.2009 18:19:00	8600966224	Dopravení pomocí PPL [120,- Kč, do 24 hod.]	Online platební kartou Typ: MasterCard	3	96 736,- Kč	10.05.2009 - 18:19:00 Objednáno z externího skladu Zboží bude k dispozici do 24 hod.
01.05.2009 18:40:04	8600171984	Dopravení pomocí Pošta [89,- Kč, do 48hod.]	Online platební kartou Typ: MasterCard	1	25 669,- Kč	05.05.2009 - 18:40:04 Bude odesláno zákazníkovi Nejpozději do 2hod.
Historie objednávek						
Datum Hodina	Variabilní symbol	Předání zboží	Platba	Celk. kusů	Cena celkem	Datum Vyřízení
21.03.2009 15:42:39	8600425636	Dopravení pomocí PPL [120,- Kč, do 24 hod.]	Online platební kartou Typ: MasterCard	3	96 736,- Kč	01.04.2009 - 18:20:55 Objednávka vyřízena

Obr. 39 Ukázka přehledu vlastních objednávek.

5.5.3.2 Objednávky k vyřízení

Jedná se o shodnou stránku s mojí objednávkou, kterou jsme si popsali v předchozím výkladu s tím rozdílem, že jsou zde k dispozici všechny objednávky uživatelů, kteří si něco v našem e-shopu objednali. Vyřizovat je může pouze redaktor nebo uživatel s vyššími právy. Navíc je zde možnost na výběr stavu, který je pro daný stav objednávky aktuální. Při změně stavu systém automaticky odešle uživateli e-mail s informací o dané změně. Tabulka je také rozšířena o údaje uživatele, aby vyřizující osoba věděla, kam zboží zaslat. V tabulce můžeme vidět všechny důležité informace o objednávce a způsobu dodání, které zákazník požaduje. Pro zjištění informace, jaké zboží zákazník požaduje, se můžeme podívat zvolením datumu nebo variabilního symbolu, pomocí něhož se zobrazí stránka se seznamem objednaného zboží.

Implementace je řešena načtením všech dat z tabulky „objednávky“. Data se z tabulky načítají pomocí SQL příkazu a následně php kódem byl naprogramován formát, jakým se mají data zobrazit, což je prováděno pomocí cyklu, který posupně načítá řádky databáze a zobrazuje je na stránce. Pokud objednávka obsahovala více notebooků, tak je v tabulce více řádků stejného data a variabilního symbolu, a zároveň tyto informace nám poslouží k zobrazení kompletní objednávky. Nastavení stavu je řešeno pomocí formuláře. Pokud dojde ke změně stavu, tak se v tabulce objednávka aktualizuje sloupec vyřízení a dvyřízení, kam se uloží datum poslední změny, které se zobrazí uživateli v kategorii moje objednávky a vyřizující osobě přímo pod nastavovacím formulářem včetně data změny.

pepa	Josef Novák	Česká 25 Brno	majkl.knight@hotmail.com 774112123	60100	Nastavit	Připraveno k převzetí
01.05.2009 18:40:04	8600171984	Dopravení pomocí Pošta [89,- Kč, do 48hod.]	Online platební kartou Typ: MasterCard	1	25 669,- Kč	05.05.2009 - 18:40:04 Bude odesláno zákazníkovi Nejpozději do 2hod.
pepa	Josef Novák	Česká 25 Brno	majkl.knight@hotmail.com 774112123	60100	Nastavit	Objednáno z externího skladu
10.05.2009 18:19:00	8600966224	Dopravení pomocí PPL [120,- Kč, do 24 hod.]	Online platební kartou Typ: MasterCard	3	96 736,- Kč	10.05.2009 - 18:19:00 Objednáno z externího skladu Zboží bude k dispozici do 24 hod.
pepa	Josef Novák	Česká 25 Brno	majkl.knight@hotmail.com 774112123	60100	Nastavit	Objednáno od výrobce
20.05.2009 00:49:38	8600993915	Dopravení pomocí PPL [120,- Kč, do 24 hod.]	Online platební kartou Typ: MasterCard	3	96 736,- Kč	20.05.2009 - 00:54:09 Objednáno od výrobce Zboží bude k dispozici do 3dnů.

Obr. 40 Ukázka systému k vyřízení objednávek.

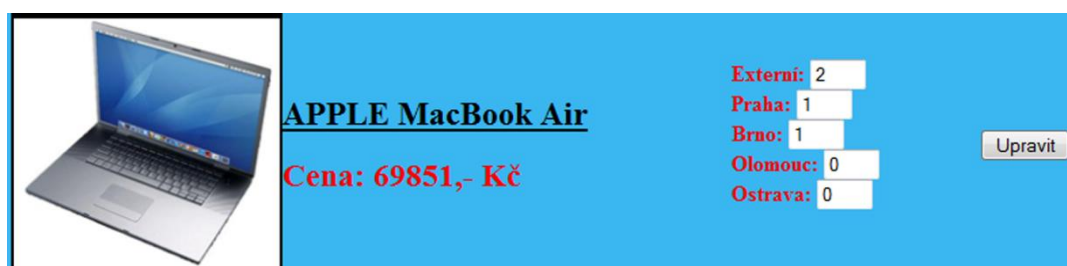
5.6 Skladovací systém

Úkolem skladovacího systému je udržovat aktuální stav notebooků ve skladech. V aplikaci je vidět přesný počet notebooků ve všech pěti skladech, které jsme si pro ilustraci navrhli po přihlášení do systému. V případě, že nejsme přihlášení, zobrazí se nám pouze informace, jestli je notebook skladem nebo ne.

Při navrhování aplikace jsme se rozhodli pro navržení čtyř klasických obchodů v různých městech a jednoho externího skladu, který slouží především k vyřizování objednávek z e-shopu a zajišťuje dodávání zboží do klasických obchodů. Tyto sklady jsme si pojmenovali následujícím způsobem tak, že klasické obchody jsou pojmenovány podle města, v kterých se nachází:

- Externí
- Praha
- Brno
- Olomouc
- Ostrava

Informace o skladech je viditelná ve všech částech katalogu notebooků, i ve vyhledávači. U přehledů v katalogu se na pravé straně vedle tlačítka pro přidání do košíku zobrazuje informace o tom, jestli notebook je na skladě. U detailního popisu je informace o množství notebooků vidět nad obrázkem. Informace o skladech si můžeme prohlédnout na Obr. 21, 30 a 45, kde je vidět informace pro nepřihlášené uživatele a dále na Obr. 31, 36 a 43, kde je vidět informace pro přihlášené osoby. K editaci skladů slouží formulář Obr. 41



APPLE MacBook Air
Cena: 69851,- Kč

Externí: 2
Praha: 1
Brno: 1
Olomouc: 0
Ostrava: 0

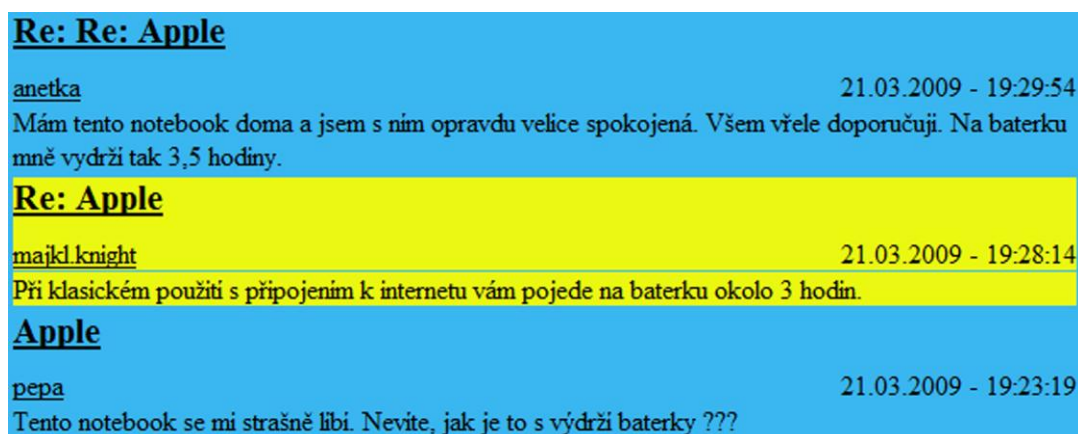
Upravit

Obr. 41 Ukázka správy skladovacího systému.

5.7 Diskusní systém

Pro diskusi mezi uživateli, a zároveň i se zaměstnanci obchodního řetězce, byl navržen diskusní systém, který je připojen ke každému notebooku v katalogu. U každého notebooku se tedy na konci stránky dozvíme, jestli pro daný notebook existuje diskuse a pokud jsme přihlášení, tak nám umožní diskusní systém číst a také přispívat. V opačném případě uvidíme jen, jestli existuje k danému notebooku diskuse a v případě, že existuje, tak uvidíme jen počet příspěvků. V detailním zobrazení notebooků jsou viditelné vždy jen poslední dva příspěvky a také odkaz na vstup do diskuse, kde si můžeme přečíst celou diskusi k vybranému notebooku, anebo vložit nový příspěvek.

Implementace diskuse byla navržena tak, že id notebooku je rovno id fóra. Pro vkládání příspěvku byl navrhnut JavaScriptový formulář, který je při psaní příspěvku schopný zobrazovat, kolik ještě můžeme napsat znaků, než dosáhneme maximálního limitu, tj. 1000 znaků, které souvisí s omezením databáze, kde je navrženo, že obsah zprávy může mít pouze stanovený počet znaků. Uprostřed pod formulářem se zobrazuje červený pruh, který se zvětšuje a jakmile dosáhneme hranice formuláře, tak se zobrazí upozornění o tom, že bylo dosaženo maximálního limitu počtu znaků. Příspěvky uživatelů, kteří mají práva redaktora a vyšší, se zobrazují se žlutým podkladem, aby uživatel rychle rozeznal odpověď někoho z obchodu (viz. Obr. 42).



Obr. 42 Ukázka diskuse k notebooku.

Na předchozím obrázku je diskuse k notebooku Apple Macbook Air. Uživatelé, s právy redaktor a vyššími, mají k dispozici i souhrn diskuse pro rychlejší pročtení, o čem se diskutuje a v případě položení otázky i možnost odpovědět nebo nevhodné příspěvky smazat.

6 ZÁVĚR

Úkolem diplomové práce bylo navrhnout a implementovat aplikaci na zpracování neurčitých údajů v databázích. Při stanovování požadavků, které by měla aplikace splňovat, bylo rozhodnuto, že neurčité údaje budou představovat požadavky zákazníků při nákupu zboží. Jako základní zboží pro nákup byly vybrány notebooky, protože v dnešní době jsou velmi oblíbené a lidé si je stále častěji pořizují z důvodů jejich mobility.

V rámci diplomové práce bylo vytvořit internetový obchod s expertním znalostním systémem, který má za úkol poradit zákazníkům, kteří se v parametrech notebooků neorientují. Pro ostatní byl implementován klasický katalog, jaký známe z různých obchodů na internetu, které se také zabývají prodejem výpočetní techniky.

V úvodní části práce byly popsány výhody využití fuzzy logiky s příklady úspěšných aplikací využívající tuto logiku. V teoretické části byly zejména popsány základy fuzzy logiky, fuzzy čísla, operace s fuzzy množinami včetně lingvistických proměnných a evaluačních výrazů přirozeného jazyka, které jsou potřebné pro realizaci aplikace. V druhé části byla zmíněna historie internetu a moderních programovacích technik k vývoji dynamických internetových aplikací. Všechny výše zmíněné teoretické aspekty byly využity v praktické části diplomové práce, jejímž výsledkem je znalostní expertní fuzzy systém, na který byl kladen zejména požadavek na jednoduchost a snadnost modelování fuzzy modelu, na základě něhož se bude expertní systém dále rozhodovat. Díky tomuto systému jsme schopni nabídnout zákazníkovi popis konkrétních technických parametrů notebooků a současných trendů ve výpočetní technice pomocí neurčitých výrazů přirozeného jazyka, který je základním dorozumívacím prostředkem mezi lidmi. Pomocí neurčitých výrazů tedy zákazníkovi vyhledáme nejvýhodnější notebook dle jeho požadavků, které sdělují prodáváci při nákupu zboží v klasickém obchodě. Tím se aplikace přibližuje klasickému nákupu v obchodě. Zbytek textu řeší zejména popis všech součástí internetového obchodu, mezi které patří katalog zboží, virtuální košík, objednávací a expediční systém, skladovací systém a v neposlední řadě diskusní systém. U všech výše zmíněných součástí aplikace je v textu řešena zejména implementace, návrh datových struktur, popis funkcí a to vše je doplněno o ilustrační obrázky právě popisované části aplikace.

Při vývoji aplikace byl především kladen požadavek na jednoduchost, přehlednost a intuitivní ovladatelnost jednak pro uživatele, tak i pro experty, kteří budou spravovat model našeho expertního fuzzy systému. Vzhled aplikace byl navržen na míru našeho internetového obchodu se znalostním expertním fuzzy systémem.

Protože internet se stále rozrůstá a každým dnem se k němu připojují noví uživatelé na celém světě, lze jej řadit mezi nejoblíbenější informační a komunikační prostředek mezi lidmi. Se zvyšujícím trendem nakupování na internetu by námi navržený e-shop s expertním fuzzy systémem mohl být implementován ve všech obchodech. Důvod, proč takové aplikace nejsou již dnes součástí internetových obchodů, je dle mého názoru zejména ten, že programátor, který obdrží zakázku vyhotovení e-shopu, neví, že by mohl prodejci nabídnout implementaci takového systému internetového obchodu, který by uměl zákazníkům e-shopu také poradit s nákupem zboží. Další možností může být to, že naprogramování znalostního expertního systému k moderní aplikaci e-shopu zvýší cenu,

kterou musí prodejce za ni zaplatit, což je také podstatným aspektem při objednávání podobné internetové aplikace.

Poněvadž jde o e-shop, kde internet je multiplatformní, tudíž nezávislý na operačním systému počítače, nemá naše aplikace žádné omezení. Avšak než se aplikace e-shopu s expertním fuzzy systémem nasadí do ostrého provozu, bylo by vhodné ještě uvážit možnosti implementace vyhledávacího algoritmu na principu neurčitých lingvistických požadavků od zákazníka zahrnující další (v odborné literatuře popsané) metody vícekritériálního rozhodování, než jsme použili v naší aplikaci a tyto porovnat. Domníváme se ale, že použitý rozhodovací algoritmus je pro naše účely dostačující a zákazníkům je schopen poskytnout kvalifikovanou odpověď na jejich požadavky.

V případné návazné diplomové práci by bylo pak možné se ještě zaměřit na implementaci plateb a to především platby pomocí online platební karty, kde by bylo potřeba si následující funkci domluvit s bankou a platbu pomocí úvěru by bylo také nutné domluvit s nějakou úvěrovou společností. Vyřešení těchto úkolů, zejména ohledně plateb za zboží, umožní univerzální využití aplikace a její nasazení do ostrého provozu.

7 SEZNAM LITERATURY

- [1] NOVÁK, V.: *Základy fuzzy modelování*. Praha: BEN – technická literatura, 2003. ISBN 80-7300-009-1.
- [2] MODRLÁK, O.: *Fuzzy řízení a regulace*. Elektronický studijní materiál, Univerzita Liberec, 2004, http://www.fm.vslib.cz/~krtsub/fm/modrlak/pdf/tar2_fuz.pdf.
- [3] WELLING, L. – THOMSON, L.: *PHP a MySQL rozvoj webových aplikací*. Praha: SoftPress, 2004. ISBN 80-86497-83-6.
- [4] LACKO, L.: *Ajax Hotová řešení*. Brno: Computer Press, a.s., 2008. ISBN 978-80-251-2108-5.
- [5] ŠPINAR, D.: *Tvoříme přístupné webové stránky*. Brno: Zoner Press, 2004. ISBN 80-86815-11-0.
- [6] ŠEDA, M.: *Database systems*. Elektronický studijní materiál, VUT v Brně, FSI, 2005, http://www.uai.fme.vutbr.cz/~mseda/DBS05_eng.pdf.
- [7] ROUPEC, J.: *Počítačové sítě*. Elektronický studijní materiál. VUT v Brně, FSI, listopad 2002, http://drogo.fme.vutbr.cz/~jroupec/nsite/p_site.pdf.
- [8] DVOŘÁK, J.: *Expertní systémy*. Elektronický studijní materiál, VUT v Brně, FSI, 2004, <http://www.zam.fme.vutbr.cz/~jdvorak/Opory/ExpertniSystemy.pdf>.
- [9] MORÁVEK, P.: *WWW aplikace popisující plány budov FSI*. Bakalářská práce, VUT v Brně, FSI, 2007, 36 s.
- [10] WIKIPEDIA, otevřená encyklopedie. Poslední revise 27.03.2009, <http://www.wikipedia.org>.
- [11] LERDORF, R., TATROE, K., MacIntyre P.: *Programming PHP*. Sebastopol: O'Reilly Media, Inc., 2006. ISBN 978-0596006815.
- [12] DUBIOS, P.: *MySQL*. Addison-Wesley, 2008. ISBN 978-0672329388.
- [13] ZAKAS, C. N., MCPEAK, J., FAWCETT, J.: *Professional Ajax*. Wrox, 2007. ISBN 978-0470109496
- [14] NQUYEN, T. H., WALKER, A. E.: *A First Course in Fuzzy Logic*. Chapman & Hall/CRC, 2005. ISBN 978-1584885269.
- [15] GALINDO, J.: *Fuzzy Databases: Modeling, Design and Implementation*. Idea Group Publishing, 2005. ISBN 978-1591403241.
- [16] GALINDO, J.: *Handbook of Research on Fuzzy Information Processing in Databases*. Information Science Reference, 2008. ISBN 978-1599048536.
- [17] FLANAGAN, D.: *JavaScript: The Definitive Guide*. Sebastopol: O'Reilly Media, Inc., 2006. ISBN 978-0596101992.
- [18] EHRGOTT, M.: *Multicriteria Optimization*. Berlin: Springer, 2005. ISBN 978-3540213987.

8 SEZNAM OBRÁZKŮ

Obr. 1 Skoková funkce klasické množiny [2].....	15
Obr. 2 Průběh funkce fuzzy množiny [2].	15
Obr. 3 Funkce příslušnosti fuzzy čísla [1].	18
Obr. 4 Minimální funkce [1].....	18
Obr. 5 Střední funkce [1].	18
Obr. 6 Maximální funkce [1].	19
Obr. 7 Funkce příslušnosti [2].	20
Obr. 8 Základní evaluační trichotomie [1].....	21
Obr. 9 Tvary fuzzy množin [1].	22
Obr. 10 Obdélníkový tvar [1].	22
Obr. 11 Tvary zjednodušených jazykových proměnných [1].....	23
Obr. 12 Relace mezi tabulkami.....	34
Obr. 13 Obrazovka nastavení lingvistických proměnných pro dané parametry.....	36
Obr. 14 Výběr funkce fuzzy množin.	38
Obr. 15 Vložení nové lingvistické proměnné.	38
Obr. 16 Model lingvistických proměnných pro velikosti notebooků.	39
Obr. 17 Výběr lingvistické proměnné u generování fuzzy ohodnocení.	40
Obr. 18 Výpočet fuzzy hodnot.....	41
Obr. 19 Tabulka notebooků a jejich fuzzy ohodnocení.	42
Obr. 20 Vyhledávací formulář pro expertní systém.	45
Obr. 21 Vyhledávací dotaz a výsledek jeho vyhledávání.	45
Obr. 22 Relace tabulek e-shopu.	52
Obr. 23 Relace všech tabulek v databázi naší aplikace.	53
Obr. 24 Nepřihlášený uživatel.	53
Obr. 25 Přihlášený uživatel.....	53
Obr. 26 Formulář k registraci do aplikace.	54
Obr. 27 Ukázka potvrzovacího e-mailu.	55
Obr. 28 Ukázka možnosti nastavení uživatelského účtu v aplikaci.....	55
Obr. 29 Ukázka administrace práv uživatele SuperAdmin.....	57
Obr. 30 Ukázka přehledu katalogu notebooků.	58

Obr. 31 Podrobný popis notebooků v katalogu.....	58
Obr. 32 Formulář k vložení nového a editaci notebooku.....	59
Obr. 33 Kategorie v katalogu pro uživatele.	60
Obr. 34 Kategorie v katalogu pro odborníky.	60
Obr. 35 Ukázka informací o objednatelce.	61
Obr. 36 Ukázka obsahu košíku.	61
Obr. 37 Ukázka prvního kroku dokončení objednávky.	62
Obr. 38 Ukázka dokončení objednávky a platby pomocí platební karty.	64
Obr. 39 Ukázka přehledu vlastních objednávek.	65
Obr. 40 Ukázka systému k vyřízení objednávek.....	66
Obr. 41 Ukázka správy skladovacího systému.	66
Obr. 42 Ukázka diskuse k notebooku.	67
Obr. 43 Ukázka stránky aplikace.	75
Obr. 44 Model lingvistických proměnných pro velikost notebooků.	76
Obr. 45 Vyhledávací dotaz a výsledek jeho vyhledávání.	77
Obr. 46 Kompletní relace tabulek v databázi.....	78

9 PŘÍLOHY

9.1 Ukázka stránky aplikace

FUZZY DATABÁZE NOTEBOOKU

Bc. Petr Morávek

VUT Brno, FSI, ÚAI

Seznam notebooků:

Kvalita notebooku: **Vše** | [nejkvalitnější](#) | [střední](#) | [levnější](#) | [kvalitní](#) | [neutrální](#) |

Velikost notebooku: **Vše** | [největší](#) | [normální](#) | [menší](#) | [nejmenší](#) | [větší](#) |

Cena: [Nízká](#) | [Vysoká](#)

Odborník | **Uživatel**

1 2 3 4 5 6 7 8 >

	<p><u>Acer Aspire one A110-Ab</u></p> <p>Intel Atom N270 1.6 GHz, paměť DDR2 1 GB, pevný disk 8 GB (4200 ot./min), velikost displeje 8.9", rozlišení displeje 1024 x 600, grafická karta integrovaná, síťová karta, wifi karta, SD, 1.1 kg</p> <p>Cena: 6 075,- Kč</p> <p style="text-align: right;">Není skladem</p>
	<p><u>Asus Eee PC 900A</u></p> <p>Intel Atom N270 1.6 GHz, paměť DDR2 1 GB, pevný disk 16 GB (4800 ot./min), velikost displeje 8.9", rozlišení displeje 1024 x 600, grafická karta Intel® Graphics Media Accelerator, síťová karta, wifi karta, SD, 0.99 kg</p> <p>Cena: 7 055,- Kč</p> <p style="text-align: right;">Skladem: 4ks</p>
	<p><u>Asus X58C</u></p> <p>Intel Celeron M 220 1.2 GHz, paměť DDR2 0.512 GB, pevný disk 160 GB (5400 ot./min), velikost displeje 15.4", rozlišení displeje 1280x800, grafická karta integrovaná, síťová karta, wifi karta, SD, MMC, MS, MS-Pro, mini SD, 2.8 kg</p> <p>Cena: 8 985,- Kč</p> <p style="text-align: right;">Skladem: 3ks</p>
	<p><u>Asus Eee PC 1000HD</u></p> <p>Intel Celeron M 0.9 GHz, paměť DDR2 1 GB, pevný disk 160 GB (5400 ot./min), velikost displeje 10", rozlišení displeje 1024 x 600, grafická karta integrovaná, síťová karta, wifi karta, bluetooth, SD, 1.45 kg</p> <p>Cena: 9 170,- Kč</p> <p style="text-align: right;">Skladem: 2ks</p>
	<p><u>Asus Eee PC 1000H</u></p> <p>Intel Atom N270 1.6 GHz, paměť DDR2 1 GB, pevný disk 160 GB (4800 ot./min), velikost displeje 10", rozlišení displeje 1024 x 600, grafická karta Intel® Graphics Media Accelerator, síťová karta, wifi karta, bluetooth, SD/MMC, 1.45 kg</p> <p>Cena: 9 595,- Kč</p> <p style="text-align: right;">Skladem: 4ks</p>
	<p><u>Umax VisionBook M760S</u></p> <p>Intel Pentium Dual-Core Mobile T2390 1.86 GHz, paměť DDR2 1 GB, pevný disk 160 GB (5400 ot./min), velikost displeje 15.4", rozlišení displeje 1280 x 800, grafická karta integrovaná, síťová karta, wifi karta, (MS/MS Pro/SD/MMC, 2.5 kg</p> <p>Cena: 11 196,- Kč</p> <p style="text-align: right;">Skladem: 8ks</p>

[super \[SuperAdmin \]](#)
[Odhlásit se](#) | [Nastavení](#)

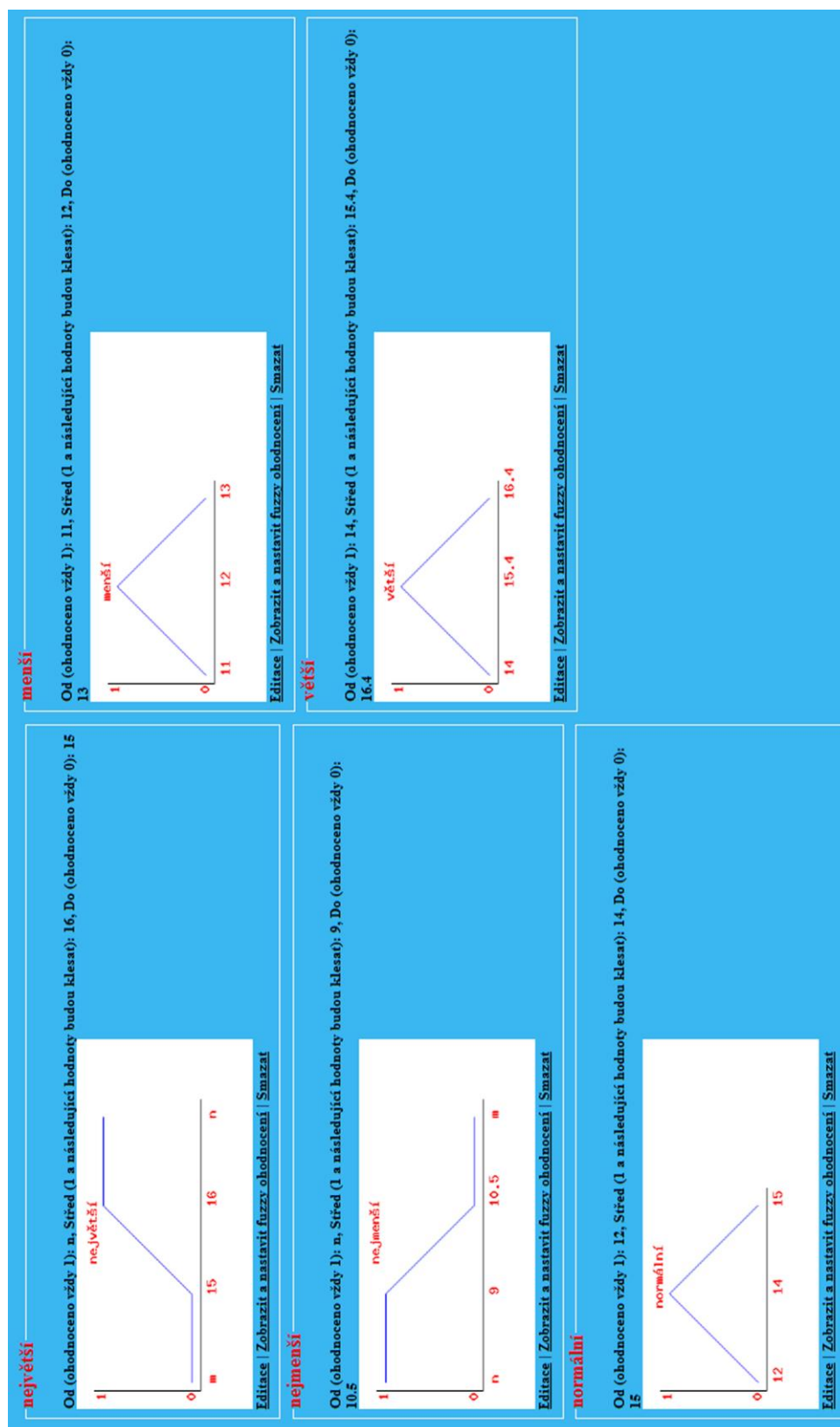
V košíku máte 0 notebooku.
Celková cena: 0,- Kč

Uživatel

[Home](#)
[Novinky v katalogu](#)
[Katalog notebooků](#)
[Uživatel](#)
[nejkvalitnější](#)
[střední](#)
[levnější](#)
[kvalitní](#)
[neutrální](#)
[Odborník](#)
[Moje objednávky](#)
[Přidat notebook](#)
[Objednávky k vyřízení](#)
[Sklady notebooků](#)
[Zobrazit fórum](#)
[Nastavení](#)
[lingvistických proměnných](#)
[Generování fuzzy proměnných](#)
[Nastavení práv](#)
[Nastavení typů](#)
[dopravy](#)
[Historie vývoje aplikace](#)

Obr. 43 Ukázka stránky aplikace.

9.2 Model lingvistických proměnných



Obr. 44 Model lingvistických proměnných pro velikost notebooků.

9.3 Vyhledávací dotaz a výsledek jeho vyhledávání

Kvalita značky notebooku: nejkvalitnější

Počítač na: práci

Velikost notebooku: menší

Rychlost notebooku: nejrychlejší

Velikost disku: hodně místa

Počet spuštěných programů: nejvíce

Výdrž baterie notebooku: velká

Hmotnost notebooku: lehký

Cena notebooku: střední

Vyhledat ideální notebook

Vašemu zadání vyhovují následující notebooky:

Asus F9E

Intel Core2 Duo T8100 2.1 GHz, paměť DDR2 2 GB, pevný disk 250 GB (5400 ot./min), velikost displeje 12.1", rozlišení displeje 1280x800, grafická karta integrovaná, síťová karta, wifi karta, bluetooth, SD, MMC, MS, MS-Pro, xD, mini SD, 2 kg

Cena: **19 590,- Kč**

Sony VAIO CS21S/R

Intel® Core™ 2 Duo T6400 2 GHz, paměť DDR2 4 GB, pevný disk 320 GB (5400 ot./min), velikost displeje 14.1", rozlišení displeje 1280 x 800, grafická karta NVIDIA® GeForce® 9300M GS, síťová karta, wifi karta, bluetooth, SD, MMC, MS, MS-Pro, xD, mini SD, 2.6 kg

Cena: **26 880,- Kč**

Acer Aspire 2930

Intel® Core2 Duo™ P8400 2.26 GHz, paměť DDR2 4 GB, pevný disk 320 GB (5400 ot./min), velikost displeje 12.1", rozlišení displeje 1280 x 800, grafická karta Intel® Graphics Media Accelerator, síťová karta, wifi karta, bluetooth, SD, MMC, MS, MS PRO, xD, 2.1 kg

Cena: **20 420,- Kč**

Umax VisionBook TN121R

Intel Core 2 Duo T8300 2 GHz, paměť DDR2 2 GB, pevný disk 320 GB (5400 ot./min), velikost displeje 12.1", rozlišení displeje 1280 x 800, grafická karta integrovaná, síťová karta, wifi karta, bluetooth, MS, MS Pro, SD, MMC, 2 kg

Cena: **26 393,- Kč**

FSC U9200

Intel Pentium Dual Core T3200 2 GHz, paměť DDR2 2 GB, pevný disk 250 GB (5400 ot./min), velikost displeje 12.1", rozlišení displeje 1280x800, grafická karta Intel® Graphics Media Accelerator, síťová karta, wifi karta, bluetooth, SD/MS/xD, 1.8 kg

Cena: **17 960,- Kč**

DELL Vostro 1310

Intel Core2 Duo T9300 2.5 GHz, paměť DDR2 2 GB, pevný disk 320 GB (5400 ot./min), velikost displeje 13.3", rozlišení displeje 1280x800, grafická karta Intel Graphics Media Accelerator, síťová karta, wifi karta, bluetooth, SD, MMC, 2.02 kg

Cena: **29 430,- Kč**

DELL XPS 1330

Intel® Core2 Duo™ T8100 2.1 GHz, paměť DDR2 2 GB, pevný disk 320 GB (5400 ot./min), velikost displeje 13.3", rozlišení displeje 1280x800, grafická karta NVIDIA® GeForce® 8400M GS, síťová karta, wifi karta, bluetooth, SD, MMC, 1.79 kg

Cena: **31 290,- Kč**

DELL Studio 1535

Intel Core2 Duo T5550 1.83 GHz, paměť DDR2 3 GB, pevný disk 320 GB (5400 ot./min), velikost displeje 15.4", rozlišení displeje 1280 x 800, grafická karta Intel Graphics Media Accelerator, síťová karta, wifi karta, bluetooth, SD/MMC, 2.77 kg

Cena: **20 525,- Kč**

APPLE MacBook Pro

Intel Core 2 Duo T9300 2.5 GHz, paměť DDR2 4 GB, pevný disk 320 GB (5400 ot./min), velikost displeje 15.4", rozlišení displeje 1920x1200, grafická karta NVIDIA GeForce 8600M, síťová karta, wifi karta, bluetooth, SD/MMC/MS/MS Pro, 2.99 kg

Cena: **64 258,- Kč**

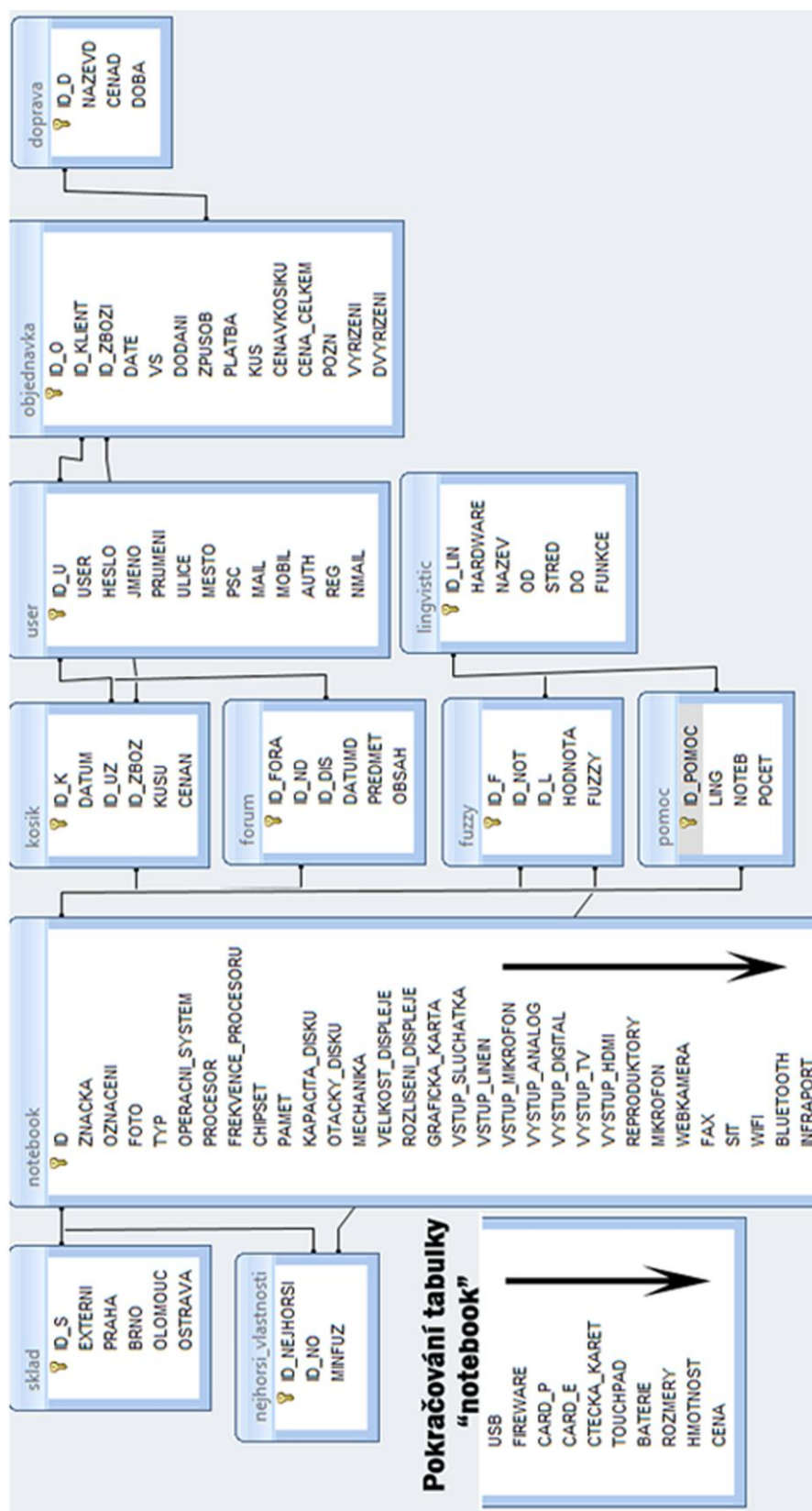
LENOVO THINKPAD X200s

Intel Core 2 Duo Processor SL9400 1.86 GHz, paměť DDR2 2 GB, pevný disk 250 GB (5400 ot./min), velikost displeje 12.1", rozlišení displeje 1280 x 800, grafická karta Intel 4500MHD, síťová karta, wifi karta, bluetooth, SD/SDHC/MMC/MS/MS Pro, 1.1 kg

Cena: **40 684,- Kč**

Obr. 45 Vyhledávací dotaz a výsledek jeho vyhledávání.

9.4 Kompletní relace tabulek v databázi



Obr. 46 Kompletní relace tabulek v databázi.